

Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»

Інститут електроенергетики

(інститут)

факультет інформаційних технологій

(факультет)

Кафедра інформаційних технологій та комп'ютерної інженерії

(повна назва)

ПОЯСНЮВАЛЬНА ЗАПИСКА

кваліфікаційної роботи ступеня бакалавра

(бакалавра, спеціаліста, магістра)

Студента Крутько Владислава Сергійовича

(ПІБ)

академічної групи 126-17-1

(шифр)

спеціальності 126 «Інформаційні системи та технології»

(код і назва спеціальності)

за освітньо-професійною програмою

«Інформаційні системи та технології»

(офіційна назва)

на тему Розробка сервісу «Календар» з функціями to-do аркуша на основі

фреймворка Django

(назва за наказом ректора)

Керівники	Прізвище, ініціали	Оцінка за шкалою		Підпис
		рейтинговою	інституційною	
кваліфікаційної роботи	проф. Коротенко Г.М.			
розділів:				

Рецензент	доц. Коротенко Л.М.			
-----------	---------------------	--	--	--

Нормоконтролер	проф. Коротенко Г.М.			
----------------	----------------------	--	--	--

Дніпро
2021

ЗАТВЕРДЖЕНО:

завідувач кафедри

інформаційних технологій
та комп'ютерної інженерії

(повна назва)

_____ Гнатушенко В.В.
(підпис) (прізвище, ініціали)

« _____ » _____ 2021 року

ЗАВДАННЯ
на кваліфікаційну роботу
ступеня бакалавр
(бакалавра, спеціаліста, магістра)

студенту _____ Крутько В.С. _____ академічної групи _____ 126-17-1
(прізвище та ініціали) (шифр)спеціальності _____ 126 « Інформаційні системи та технології »

за освітньою-професійною програмою _____

_____ «Інформаційні системи та технології»

на тему _____ Розробка сервісу “Календар” з функціями todo-аркуша на основі
_____ фреймворка Djangoзатверджену наказом ректора НТУ «Дніпровська політехніка» від 07.06.2021 р. № 317-с

Розділ	Зміст	Термін виконання
Розділ 1	Ознайомлення з матеріалами сайтів, літературою та виконання пошуку технологій і формування відповідних рішень	01.02.2021 – 30.03.2021
Розділ 2	Реалізація проектних рішень	01.04.2021 – 24.06.2021

Завдання видано _____ Коротенко Г.М.
(підпис керівника) (прізвище, ініціали)Дата видачі _____ 01.02.2021 р.Дата подання до екзаменаційної комісії _____ 24.06.2021р.Прийнято до виконання _____ Крутько В.С.
(підпис студента) (прізвище, ініціали)

Реферат

Пояснювальна записка: 92 сторінки, 17 рис., 19 табл., 4 додатки, 13 джерел.

Об'єкт розробки: Веб-додаток календар з функціями to-do аркуша.

Мета дипломної роботи: Створити засіб ефективного планування справ.

У вступі зазначена мета дипломної роботи, обґрунтована її актуальність на даний момент та конкретизовані задачі, які необхідно вирішити.

У першому розділі пояснювальної записки наведено роль календаря в історії людства, опис календарних систем. Розглянуті технології, які будуть використані в процесі розробки.

У другому розділі визначені підстави та призначення розробки, вимоги до програмної розробки, описана структура веб-додатку та протестована його робота.

Практична значність даної роботи полягає в можливості її використання користувачем для підвищення ефективності планування і моніторингу його задач та справ.

CSS, ФРЕЙМБОРК, PYTHON, КАЛЕНДАР, ВЕБ-ДОДАТОК, DJANGO, HTML.

Abstract

Explanatory note: 92 pages, 17 figures, 19 tables, 4 supplements, 13 sources.

Development object: Web application calendar with to-do worksheet functions.

The purpose of the qualification work: to create a tool for effective business planning.

The first section of the explanatory note provides the role of the calendar in human history, a description of calendar systems. The technologies that will be used in the development process are considered.

The second section defines the grounds and purposes of development, software development requirements, describes the structure of the web application and tested its operation.

The practical significance of this work lies in the possibility of its use by the user to increase the efficiency of planning and monitoring of its tasks and affairs.

CSS, FRAMEWORK, PYTHON, CALENDAR, WEB-APPLICATION, DJANGO, HTML.

ЗМІСТ

Перелік умовних позначень	7
Вступ	7-9
Розділ 1. Аналіз стану області рішення задачі. Календар. Мова програмування Python. Фреймворк Django.	
1. Календар	9-49
1.1. Історія календаря	9-11
1.2. Календарні системи	11-12
1.3. Історичні календарі	13-32
1.4. Сучасні календарі	32-49
2. Мова Python	49-54
2.1. Історія створення	49-51
2.2. Особливості Python	51-54
3. Фреймворк Django	54-59
3.1. Історія створення	54
3.2. Архітектура	55-56
3.3. Переваги та недоліки	56-59
Розділ 2. Проектні рішення. Розробка додатку за допомогою фреймворку Django.	
1. Технічне завдання для розробки web-додатку	59-62
1.1. Найменування і область застосування	59
1.2. Підстави для розробки	59
1.3. Призначення розробки	60
1.4. Вимоги до функціональних характеристик	60
1.5. Вимоги до надійності	60
1.6. Умови експлуатації	60
1.7. Вимоги до складу і параметрів технічних засобів	60
1.8. Вимоги до інформаційної та програмної сумісності	61

1.9.	Вимоги до маркування та упаковки	61
1.10.	Вимоги до транспортування та зберігання	61
1.11.	Вимоги до програмної документації	61
1.12.	Стадії та етапи розробки	61-62
1.13.	Порядок контролю та приймання	62
2.	Архітектура додатку	63-68
2.1.	Загальна структура проекту	63-64
2.2.	Анатомія папки apps	64-65
2.3.	Анатомія папки static	66
2.4.	Анатомія папки templates	66-67
2.5.	Структура бази даних	67-68
3.	Установка ПЗ, необхідного для запуску додатку	69-72
3.1.	Установка Python	69-70
3.2.	Установка Django	71-72
4.	Тестування додатку	72-77
	Висновки	77
	Список джерел	78-79
	Додаток А. Відомість матеріалів кваліфікаційної роботи	80
	Додаток Б. Текст програми	81-89
	Додаток В. Відгук керівника кваліфікаційної роботи	90
	Додаток Г. Рецензія	91
	Додаток Д. Протокол аналізу системи антиплагіат	92

Перелік умовних позначень

- ПЗ – програмне забезпечення;
- БД – база даних;
- МП – мова програмування;
- МШ – мова шаблонів;
- CSS – Каскадні таблиці стилів;

Вступ

Люди звикли жити за розкладом. Правильне планування допомагає людині ефективно справлятися з задачами. Календар, як засіб планування та організації використовується з давніх часів. І його роль, як у житті людини, так и в розвитку людського суспільства, куди більш значна, ніж здається з першого погляду.

З найдавніших часів календар виступає основою культурно-релігійного комплексу, який ми звикли називати традицією. Так, вплетені в культуру релігій, календарі мали сакральний характер. Отже, змінення календарної системи, найчастіше зв'язані зі змінами в традиції, які зазвичай є наслідком політичних перетворень. Таким чином, можна зробити наступний висновок: календар, крім своїх інших функцій, також використовується в якості засобу культурно-політичної інтеграції.

Звісно уніфікація та розповсюдження того або іншого календаря, не було лише наслідком політично-територіальної експансії домінуючих держав. Подібні процеси відбувалися и під час культурної експансії, коли традиціям предпочиталась домінуюча культура. Наприклад, так було з вавилонським місячно-сонячним календарем, який свого часу, поширився по всьому Близькому Сходу, не стільки через завоювання, скільки через культурні перетворення. Так же було з китайською календарною системою, яка все ще є домінуючою в країнах Східної Азії.

Або візьмемо для прикладу Юліанський календар, який в імперські часи поширився по всьому Середземномор'ю та більшій частині Європи. За багатьма параметрами він поступався як Вавилонсько-іудейському, так і

Греко-македонському календарю. Юліанський календар не враховував рух Місяця, чий вплив на біосферні процеси важко переоцінити.

Вважається, що точно концептуалізувати час неможливо. Деякі дотримуються теорії лінійності часу, інші ж вважають що час циклічен. Така незбагненність часу відображується і на календарі, як на культурній концепції: кожна культура осмислює концепцію часу по своєму.

Наприклад, Ведична Індія або народи мезоамерики є прикладами культури циклічного часу. Так, згідно космології ацтеків, світ протягом 2028 років створювався та знищувався декілька разів. Світи іменувалися Сонцями та керувалися богами. У Ведичній Індії же вважалося, що світ проходить через ряд епох з деякими варіаціями в рамках однієї великої, повторюваної моделі.

І хоча різні суспільства, по різному сприймають час, любий календар підкреслює циклічність часу. В основі календаря лежать природні цикли, на основі яких, час розбивається на проміжки: дні, тижні і т.д. З розвитком суспільства, розвиваються і їх календарі. З'являються більш складні розрахунки часу, та цикли на їх основі.

З математичної точки зору календар представляє собою рівняння між трьома основними астрономічними принципами:

- a) Обертання Землі навколо своєї осі (1 доба).
- b) Обертання Місяця навколо Землі (сидеричний місяць).
- c) Обертання Землі навколо Сонця (тропічний рік).

Так як ні один з цих циклів не являє собою ціле число, побудова календаря зводиться до пошуку такої системи вставки додаткових днів та місяців, яка б поєднувала в собі точність та практичність. Для зручності, календар повинен мати відносно короткий період та зрозумілу систему інтеркаляції. [1]

З розвитком технологій календар не втратив своєї актуальності, скоріш навіть навпаки, став ще більш затребуваним. Адже кількість інформації, якою людина оперує в повсякденному житті, зростає настільки стрімко, наскільки розвиваються технології. Вже створено досить багато додатків-календарів, як для ПК, так і для мобільних пристроїв, але не всі вони є ефективними інструментами планування. Адже з розвитком технологій, зростають і вимоги до функціоналу календаря. Сучасний календар повинен не тільки надавати функцію довідника, але й мати функцію зберігання важливої інформації пов'язаної з певною датою, а також мати функціонал для планування та моніторингу задач. Такий календар буде корисним інструментом для сучасної людини.

Метою даної дипломного проекту є розробка веб-додатку календаря, який буде володіти здатністю планування та моніторингу задач, що дасть йому можливість стати ефективним інструментом планування.

Розділ 1. Аналіз стану області рішення задачі. Календар. Мова програмування Python. Фреймворк Django.

Календар

Історія календаря

У свої первісні часи людство не знало рахунку часу по роках. Інформація про світ і події передавалася в міфах, переказах і генеалогія. Вік людини визначався не хронологічно, а по соціально-умовним ознакам. Однак уже тоді, завдяки спостереженням за природою, встановлювали природні одиниці виміру часу. Найменшою такою одиницею була доба (більшість народів рахувало дні, а стародавні кельти та германці - ночі). Наступною, більш великою одиницею були Місячні місяці і їх частини, відповідні фазами Місяця (хоча в ті часи не всі народи використовували рахунок за місяцями, так наприклад, рік у балійців складався тільки з діб).

У відсутність писемності, періоди часу відзначаються зарубками на дощечках, вузликами на шнурках і т.п. А сам відлік часу ведеться від довільної точки, що не має ніякого зв'язку з астрономічними явищами. Початком року вважається прихід однієї з пор року, який визначається за природними ознаками. Зміна природніх і господарських періодів, як правило, супроводжувалася календарними обрядами. Рік ділиться на сезони, які пов'язані з порами року: наступ сезону визначався за порою року. Спочатку виділяються два сезони (зима і літо або сухий і дощовий сезони для тропіків).

Потім, з плином часу і розвитком суспільства з'являється землеробство, з яким починають розрізняти три, а згодом і чотири пори року. Так, з огляду на те, що господарство має визначення початку і кінця господарських періодів за днями сонцестояння і рівнодення, виникає потреба спеціальних спостережень за Сонцем і зірками, що зіграє важливу роль в становленні природничо-наукових знань.[2]

В умовах циклічних та непередбачуваних змін природи виникають перші календарі. У Набта-Плая (територія сучасного Єгипту) приблизно 5 тис. років до н.е. напівкочовими племенами скотарів був створений, можливо, перший річне "календарне коло", початок року на якому зазначався появою зірки Сіріус. Цей календар допомагав встановити племені, коли очікувати початок і закінчення сезону дощів, який перетворював пустельний регіон в квітучу савану, придатну для випасу худоби. Приблизно в цей же час на території сучасної Німеччини був створений так званий Гозекське коло, основою відліку для якого було зимове сонцестояння.

Саме слово «Календар» йде корінням до Древнього Риму. У Древньому Римі боржники платили відсотки у дні календ – перших чисел місяця, відповідно боргова книжка називалася *Calendarium*.

У давньоєгипетському календарі рік - це проміжок часу між двома послідовними геліакічними східами Сіріуса. Причиною появи календаря стародавнього Єгипту були розливи Нілу, що відбуваються через один і той

же проміжок часу, приблизно рівний одному року. Вони губили урожай, якщо його вчасно не збирали, і приносили родючий ґрунт. В силу цих причин для успішного ведення господарства необхідно було передбачати розливи Нілу з прийнятною точністю.

Кожен народ використовував свої способи датування історичних подій. Дехто намагався вести відлік років від створення світу: євреї датували його 3761 роком до н.е., олександрійська хронологія вважала цієї датою 25 травня 5493 року до н.е. Римляни починали відлік від легендарного заснування Риму (753 рік до н.е.). Парфяне, віфіляне і Селевкідиди вели відлік років від вступу на трон першого царя, єгиптяни - з початку правління кожної наступної династії. Свій календар засновувала кожна світова релігія: за візантійським календарем, з 14 вересня 2020 року григоріанським календарем йде 7529 рік "від створення світу", в ісламському календарі - 1442 рік Хідждри, за буддійським календарем йде 2564 рік ери Нірвана, за календарем багаї - 177 рік.

Переклад з одного літочислення в інше являє певні труднощі через різної тривалості роки і через різні дати початку року в різних системах. Рахунок року з 1 січня був введений в Римі Юлієм Цезарем в 45 році до н.е. (Юліанський календар). Юліанський календар встановив середню тривалість року в 365,25 доби: звичайні роки тривали 365 днів, один раз в чотири роки (високосний рік) - 366 днів.

Розвитком юліанського календаря є григоріанський календар (новий стиль). Він був введений при папі римському Григорія XIII 15 жовтня 1582 року замість юліанського календаря (старого стилю). [3]

Календарні системи

Через те, що ні Місячний місяць, ні Сонячний рік не містять цілого числа діб, безпосередньо співвіднести між собою ці дві одиниці виміру часу неможливо. Тому вкрай складно побудувати універсальну календарну

систему, яка була б узгоджена і з видимим рухом сонця, і з видимим рухом Місяця. Найважливішим завданням при побудові календарної системи є необхідність висловити довжину місяця і року в добі і довжину року в місяцях. У спробах максимально наблизити календарний рахунок до спостережуваних природних явищ виникли три основні типи календарів - сонячні, місячні і сонячно-місячні.

Сонячні календарі. Найпоширеніша календарна система, в основу якої покладено чергування пів року, при цьому астрономічна тривалість року округлялась до 365 або 366 діб. Проблема "зайвих діб" в різних сонячних календарях вирішувалася шляхом введення подовжених років, або створенням блукаючої календарної системи, при якому новоріччя в різні роки приходиться на різні сезони. Основним видом сонячних календарів є календарні системи, побудовані на чергуванні простих і подовжених (високосних) років.

Місячні календарі. Місячні календарі засновані на періодичності чергування Місячних фаз. Для того, щоб протягом року початок кожного місяця збігався з Молодиком, непарні місяці року повинні нараховувати 29 діб, а парні - 30 діб. Іншими словами, місячний рік містить 354 діб, тобто - він приблизно на 11,25 діб коротше сонячного року. Таким чином, основним недоліком місячного календаря є неможливість узгодити його з чергуванням пів року, оскільки довжина тропічного року не ділиться без залишку на тривалість синодичного місяця. Для приведення місячного календаря у відповідність з порами року необхідно вводити додаткові, що ускладнюють рахунок літочислення, поправки.

Місячно-сонячні календарі. Місячно-сонячні календарі побудовані за складними схемами, що враховує як чергування пів року, так і чергування Місячних фаз: умовою побудови календарної системи такого типу є періодична вставка додаткових місяців в календарні роки, що налічували 12 місяців. [4]

Історичні календарі

Шумерський календар. Шумерські календарі засновані на місячно-сонячній системі. Складаються з двох півріч - Еnten ("Холод") і Емеш ("Спека") і 12 місяців. Щоб зрівняти місячний і сонячний цикли кожні кілька років вставлявся тринадцятий місяць. Достеменно невідомо коли саме починався Новий рік. Припускають, що початок нового року, за аналогією з Давнім Єгиптом, був приурочений до розливів річок в кінці березня. Крім поділу на півріччя, існувала класифікація за сезонами: сезони Ана (XII, I, II; VI, VII, VIII місяці), сезон Енліля (III, IV, V місяці), сезон Енкі (IX, X, XI місяці).

З господарських і адміністративних документів шумерів, нам відомо про календарі таких міст як: Ніппур, Ур, Лагаш, Умма, Урук (частково), Адаб. Однак більш пізні шумерські джерела коментують тільки Ніппурський календар, завдяки чому достеменно відомі обряди і події, а також значення місяців тільки цього календаря.

Перші згадки Ніппурського календаря датуються XXIV століттям до н.е., але коментарі до нього датуються або Середньовавилонським (XVI - XI століття до н.е.), або Новоассирійським (VIII - VII століття до н.е.) періодами. Якщо простежити за історією розвитку календарів в цьому регіоні, стає зрозуміла причина появи цих коментарів. Ніппурський календар, спочатку використовувався лише на території міста Ніппура, з початку II тисячоліття до н.е. стає загальнодержавним календарем Стародавньої Південної Месопотамії. Потім, за часів правління Хаммурапі та розквіту Вавилонії, замінюється вавилонським календарем, створеним на його основі: перші знаки назв Ніппурських місяців приймаються в якості логограм для позначення відповідних Вавилонських місяців. До кінця II тисячоліття до н.е. значення більшості назв Ніппурських місяців забувається, тому сучасники змушені дати короткий коментар з приводу назви кожного місяця. [5]

Нижче наведені назви та значення місяців Ніппурського календаря:

Таблиця 1.1.

Назва та опис шумерських місяців

№	Название	Значение
1	BARA-ZA-GAR	“Місяць мешканців святилищ”
2	GU-SI-SU/SA	“Місяць ходу волів”
3	SIG U-SUB-BA GAR	“Місяць поміщення цегли в цегляну форму ”
4	SU-NUMUN	“ Коли в місяць сівби свято моє на полі справляють ”
5	NENE GAR	“Місяць запалювання вогнів”
6	ITI KIN-INNIN	“Місяць служби богині Інанні”
7	ITI DU-KU	“Місяць Священного Пагорбу”
8	APIN DU A	“Місяць відпускання плуга”
9	GAN-GAN-E	“Місяць виходу Вбивці”
10	KU-SEM/AB-BA-E	“Місяць священної чистоти”
11	UD-DURU (ZIZ A/AN)	“Місяць полби”

12	SE GUR KU	“Місяць жнив”
----	-----------	---------------

Вавилонський календар. Місячно-сонячний календар.

Використовувався в Месопотамії в XVIII столітті до н.е. - I столітті. В його основі лежав Ніппурський календар. На основі вавилонського календаря були створені такі близькосхідні календарі: Єврейський, Арамейський, Сирійсько-арабський.

Спочатку II тис. До н.е. в Месопотамії загальнодержавним є Ніппурський календар. В кінці XIX століття до н.е. семітські племена амореїв після майже 3-х століть вторгнень, створюють свої царства в Месопотамії. А вже через століття ці царства об'єдналися під керівництвом царя Хаммурапі. З цього моменту починається історія вавилонського календаря. Племена амореїв принесли з собою свої локальні календарі, в період панування яких (в основному в XVIII столітті до н.е.) відбулося змішання Шумерських традицій з племінними звичаями амореїв. Потім вони були знову витіснені Ніппурським календарем, хоча на нього вже вплинули семітські календарі: під ідеограмами Ніппурських місяців ховалися семітські. Цей новий календар і став називатися вавилонським. За однією з версій він був створений за царя Хаммурапі (1793 - 1750 роки до н.е.), за іншою - його сина, царя Самсуілуне (1749 - 1712 роки до н.е.).

Нижче наведена таблиця з семітськими та ніппурськими назвами місяців, а також семітські значення місяців:

Таблиця 1.2.

Місяці Вавилонського календаря

№	Семитский месяц	Ниппурский месяц	Значение
1	ni-sa-an-nu	BARA-ZA-GAR	“Рухатися”
2	a-a-gu	GU-SI-SU/SA	“Яркий”
3	si-ma-nu	SIG U-SUB-BA GAR	“ Ідеальний час

			для діла”
4	du-u-zu	SU-NUMUN	“Посвячений Думузі”
5	a-bu	NENE GAR	“Ворожебний”
6	u-lu-lu	ITI KIN-INNIN	“Очищати”
7	tas-ri-tu	ITI DU-KU	“Початок”
8	a-ra-ah-sam-na	APIN DU A	“Жаровня”
9	ki-si-li-mi	GAN-GAN-E	--
10	te-bi-tu	KU-SEM/AB-BA-E	“Мутний”
11	sa-ba-tu	UD-DURU (ZIZ A/AN)	“Руйнування”
12	ad-da-ru	SE GUR KU	“Хмарний”

У шумерських документах є свідчення, що ще в 2400 році до н.е. існувала практика додаткового місяця, щоб підганяти традиційний місяць жнив до часу збирання врожаю. Додаткові місяці вставлялися за указом влади. У вавилонському календарі, за указом царя Хаммурапі, між місяцями улулу і ташрїту вставлявся додатковий місяць "другий улулу". Згодом почав вставлятися також "другий Аддару". [5,6]

Давньоєгипетський календар. Один з найдавніших календарів, проіснував близько чотирьох тисяч років. Походження календаря пов'язане із зіркою Сиріус: геліакічне сходження Сиріуса передує літньому сонцестоянню, яке в свою чергу, передує розливу Нілу. Так як все сільське господарство древніх єгиптян було засновано на цих розливах, таке астрономічне спостереження не залишилося без уваги. Між двома геліакічними сходами Сиріуса проходить 365,25 доби - саме таку кількість днів єгиптяни почали вважати за рік. Варто зазначити, що кожні 4 роки сезонні явища відставали на добу - адже в рік було покладено ціле число. Календар був введений 19 липня 4241 року до н.е.

Одночасно існувало два календаря, які використовувалися для різних цілей. Сонячний календар використовувався для літочислення, їм датувалися офіційні документи. Місячний календар використовувався в релігійних і побутових цілях.

За сонячним календарем рік ділився на 3 сезони по 4 місяця:

- Час повіней (ахет) – з середини липня до середини листопаду.
- Час всходів (перет) – з середини листопаду до середини березню.
- Час посухи (шему) – з середини березню до середини липня.

Назви сезонів пов'язані сільськогосподарськими роботами. Місяці сонячного календаря не мали своїх унікальних назв, а просто позначалися комбінацією номеру і сезону. З огляду на те, що кожен місяць мав по 30 днів, в сумі виходило 360 днів. Відсутні 5 днів додавали до останнього місяця. Такі дні називалися Hrj.w-rnp.t (херіу-ренпет) - "знаходяться над роком". Вони вважалися днями народження наступних богів відповідно: Осіріса, Гора, Сета, Ісіді і Нефтіди. [7]

Місяці місячного календаря, на відміну від місяців сонячного, мали свої унікальні назви та називалися по релігійному святу, яке святкували в наступному місяці.

Нижче приведені названня місяців лунного календаря в різні епохи:

Таблиця 1.3.

Місяці давньоєгипетського календаря

№	Середнє царство	Нове царство	Сучасний календар
1	Техі ("Тот")	Джехуті	Липень – Серпень

2	Менхет (“вбрання”)	Паніпет	Серпень – Вересень
3	Хут-Херу (“подорож Хатор”)	Хут-Херу	Вересень – Жовтень
4	Нехеб-Кау	Ка-хер-Ка	Жовтень – Листопад
5	Шеф-Бедет (“набухання пшениці”)	Та-Аабет	Листопад – Грудень
6	Рекех-Вер (“велика пожежа”)	(Па-Эн-Па) Мехеру	Грудень – Січень
7	Рекех-Неджес (“мала пожежа”)	(Па-Эн) Имен- Хетеп	Січень – Лютий
8	Ренен-Утет (Ренутет “годуєча”)	(Па-Эн) Ренен- Утет	Лютий – Березень
9	Хенсу (Хонсу)	(Па-Эн) Хенсу	Березень – Квітень
10	Хенет-Хети-Перті	(Па-Эн) інт	Квітень – Травень
11	Іпет-Хемет	Іпі-Іпі	Травень – Червень
12	Упет-Ренпет (“відкриття року”)	Месут-Ра	Червень – Липень

Ацтекський календар. Структуру ацтекського календаря використовувало більшість мезоамериканських календарів. В його основі лежать два взаємопов'язаних цикли: *тональпоуаллі* ("рахунок днів") складався з 260 днів і *шіупоуаллі* ("рахунок років") складався з 365 днів. Шляхом накладання цих двох циклів формувався 52-річний календарний

цикл *шіумольпіллі* ("Століття") або 104-річний подвійний календарний цикл *уеуелістлі* які є основними структурними одиницями їх історії.

Тональпоуаллі формувався з комбінації двох менших, базових циклів. Першим була послідовність чисел від 1 до 13. Цифри фігурували тут не в математичному сенсі, а в якості міток, так як кожен день був пов'язаний з одним з 13-ти Владик дня.

Нижче наведені числа днів та їх боги:

Таблиця 1.4.

Числа днів *Тональпоуаллі*:

1. Шіутекутлі	Владика бірюзи
2. Тлалтекутлі	Владика землі
3. Чальчіуітлікуэ	“Вона в спідниці з нефриту”
4. Тонатіу	З його приходом настає літо
5.Тласольтеотль	Богиня бруду
6. Міктлантекутлі	Владыка Міктлана
7. Сінтеотль	Вухо бога кукурудзи
8. Тлалок	Тягнеться над землею
9. Кецалькоатль	Пернатий змії
10. Тескатліпока	Курящеся дзеркало
11. Чальмекатекутлі	Владика, котрий живе в Чальмекане
12. Тлауіскальпантекутлі	Владика зорі
13. Сітлалікуэ	“Вона в спідниці з зірок ”

Другий базовий цикл складався з 20-ти знаків і не зв'язувався з божествами безпосередньо.

Нижче наведені ці знаки:

Таблиця 1.5.

Значення знаків

Сіпактлі	Крокодил
Екатль	Вітер
Каллі	Будинок
Куэцалін	Ящірка
Коатль	Змія
Мікістлі	Смерть
Масатль	Олень
Точтлі	Кролик
Іцкуінтлі	Собака
Осоматлі	Мавпа
Маліналлі	Трава
Акатль	Очерет
Оцелотль	Ягуар
Куаутлі	Орел
Коцкакуаутлі	Гриф
Оллін	Рух (землетрус)
Текпатль	Кремій
Куіаутль	Дощ
Шочітль	Квітка

Шляхом об'єднання цих двох базових циклів позначалися дні в *тональноуаллі*. Наприклад: цикл починається з номера 1 і знака Коатль, щодня ця комбінація буде змінюватися відповідно до послідовності, поки не

дійде до 13 Текпатль. Потім послідовність знаків продовжиться, а відлік чисел почнеться з 1. Таким чином, наступним днем, після 13 Текпатль буде 1 Куіаутль. Всього таких унікальних комбінацій 260, з чого і складається рік в *тональпоуаллі*.

Цикл *тональпоуаллі* використовувався для ритуалів.

В основі другого важливого календарного циклу - *шіупоуаллі* лежав сонячний рік *шіуітль*. Він ділився на 18 місяців по 20 днів. На відміну від *тональпоуаллі*, у фундаментальній основі *шіупоуаллі* лежав 20-денний цикл. Кожен день місяця комбінувався зі знаком, як і в *тональпоуаллі*. Однак, так як чисел місяця і знаків днів рівну кількість, кожен місяць має однакову комбінацію чисел і знаків. Так як за основу *шіупоуаллі* узяли 20-ти знаковий цикл, який не має прямих асоціацій з божествами, він є більш світським. Так як, в результаті число днів дорівнює 360, щоб відносно зрівняти *шіупоуаллі* з сонячним роком, додаються 5 додаткових днів немонтемі. Вони не включалися ні в один звичайний місяць і не мали назви в *шіупоуаллі*. Однак, так як, *шіупоуаллі* і *тональпоуаллі* йшли паралельно один одному, кожен день немонтепі мав свою назву в *тональпоуаллі*. Оскільки, сонячний рік не ділиться без залишку на 20, ці 5 додаткових днів зміщували початок кожного нового *шіуітля* на 5 днів.

Шіупоуаллі використовували для визначення дат проведення обрядів і церемоній, пов'язаних з сільським господарством.

Нижче наведена таблиця назв місяців, божеств покровителів та ритуалів:

Таблиця 1.6.

Назва місяців *шіупоуаллі*

№	Назва місяця	Божество-покровитель	Ритуали
1	Атлькауало (“припинення	Тлалок,	Жертвоприношення

	води”)	Чальчіутлікуэ	дітей водняним божествам
2	Тлакашіпеуаліцтілі (“білування людей”)	Шіпе-Тотек	Жертвоприношення воїнів; танець жерця з шкірою, здертої з жертв
3	Тосостонтлі (“мале бдіння”)	Коатлікуэ, Тлалок	Поховання шкіри жертв, жертвоприношення дітей
4	Уэйтосостлі (“велике бдіння”)	Сентеотль, Чікомекакоатль	Благословення нової кукурудзи; жертвоприношення дівчини
5	Тошкатль (“посуха”)	Тескатліпока, Уіцілопочтілі	Жертвопринесення персоніфікацій богів
6	Эцалькуалістлі (“блюдо з кукурудзи і квасолі ”)	Тлалок	Ритуальне удушення персоніфікацій водних божеств; ритуальні обмивання і танці
7	Текуілуітонтілі (“маленький бенкет панів”)	Шуішточіуатль, Шочіпіллі	Жертвопринесення персоніфікацій богів; церемонія соледобувачів
8	Уэйтекуіутль (“великий	Шілонен	Свято богині

	бенкет панів”)		молодої кукурудзи; знать роздає подарунки та їжу черні
9	Тлашочімако (“народження квітів”)	Уцілопочтлі	Всі зображення богів прикрашаються квітковими гірляндами. Бенкет з кукурудзяними коржиками і індичкою
10	Шокотлуэці (“падіння плодів ”) Уэйміккаіутль (“великий бенкет мертвих ”)	Шіутекутлі	Ритуальне змагання; жертвоприношення богам вогню
11	Очпаністлі (“підмітання дороги”)	Тласольтеотль	Підмітання будинків і доріг; імітація бою
12	Теотлеко (“повернення богів”)	Тескатліпока	Привітальні церемонії на честь повернення богів на землю; ритуальне сп'яніння, вогненні жертвопринесення
13	Тепеіутль (“бенкет пагорбів”)	Тлалок	Церемонії, присвячені богам гірських дощів;

			людські жертвоприношення і ритуальний канібалізм
14	Кечоллі (“дорогоцінне перо”)	Мішкоатль- Камаштлі	Піст і подальша ритуальна полювання; жертвоприношення видобутку і ритуальний бенкет
15	Панкецалістлі (“підняття прапора”)	Уцілопочтлі	Будинки і плодові дерева прикрашаються паперовими прапорами; процесія; масові жертвоприношення
16	Атемостлі (“сходження вод”)	Тлалок	Свято на честь водних богів; жертвоприношення дітей і рабів
17	Тітітль (“витягування”)	Іламатекутлі	Магічні ритуали для викликання дощу; побиття жінок мішками з соломою, що б ті плакали
18	Іскаллі (“воскресіння”)	Шіутекутлі	Спорудження фігур богів з

			амарантового тіста; бенкет з тамалес і рослинами
	Немонтемі("порожні дні")	П'ять днів посту і стриманості без ритуалів	

Точно відомо, що рік називався по знаку дня, але існують дискусії, чи повинен був цей знак дня випадати на перший або останній день першого місяця. Наприклад, якщо 1 рік починався зі знака Точтлі, такий рік називався 1 Точтлі, а наступний за ним 2 Акатль. Таке зміщення знаків дня обумовлено вставкою немонтемі. До того ж, 13-ти денний цикл укладається в сонячний рік 28 разів в результаті даючи суму в 364 дня. Це означає, що 13-ти денний цикл починається на день раніше, ніж закінчується сонячний рік. Виходить, що на початок року доводиться вже не число 1, а 2.

Таким чином, *шіупоуаллі* і *тональпоуаллі* йдуть безперервно, формуючи більший цикл в 52 сонячних роки (включає 52 цикли *шіупоуаллі* і 73 цикли *тональпоуаллі*) в цій точці обидва циклу одночасно закінчуються, і почнуться спочатку. Це ознаменує кінець часу і початок нового календарного кола, в честь чого буде проведена церемонія Нового Вогню. [8]

Давньогрецький календар. Про ранній календар стародавніх греків нічого не відомо. З праць Гесіода і Гомера можна припускати про наявність у стародавніх греків якогось примітивного календаря, швидше за все місячного. Так, в своїх працях Гесіод перераховує всі дні протягом місяця, згадує серединні дні, а також рахує проміжок між "прибуванням" і "спаданням" місяця. Вважається, що таких "протокалендарей" було чотири: еолічеський, північно-доричний, південно-доричний, іонічний. На основі цих календарів, для кожного календаря в своєму регіоні, розвивалися локальні календарі.

Те, що прийнято вважати давньогрецьким календарем, це більш розвинений варіант цих календарів, заснований вже на місячно-сонячній системі. Достеменно невідомо, коли саме почалася використовуватися така система. Однак існує припущення, що ця календарна реформа відбулася в Дельфах у VIII столітті до н.е., правда ні підтвердити, ні спростувати це припущення не представляється можливим.

Кожен Поліс використовував власний календар, з власними назвами місяців, які зазвичай походили з назв свят, які відбувалися в цьому місяці. Початок року також був різним. Однак структура календаря була незмінною. Календар складався з 12 синодичних місяців. Місяць повинен був починатися в Молодик, проте це не завжди відбувалося точно через наступну причину.

Що б вирішити проблему різної тривалості видимих періодів обертання Місяця тривалість місяця була округлена до 29.5 днів, таким чином два місяці становили 59 днів. Таке округлення застосовувалося для ділового і громадського життя, а такі місяці іменувалися "цивільними". Цивільні місяці вважалися повними (30 днів), або порожніми (29 днів). Таким чином, місячний цикл перестав бути синхронізований з Місяцем. Тепер грекам доводилося розрізняти "громадянський" Молодик (1-й день місяця) і справжній Молодик.

У розвитку давньогрецького календаря виділяють, як мінімум, три етапи:

- До Солона

Місяць ділиться на три декади, рахунок ведеться по днях в декаді. Рік починається під час зимового сонцестояння, потім початок року переноситься на літнє сонцестояння. Початком доби вважається вечір. Додатковий місяць вставляється в міру необхідності. Рік складається з 12 місяців по 29 і 30 днів, що в сумі дає 354 дня.

- Після Солона (приблизно з 590 року до н.е.)

У міру впорядкування календаря, був введений 8-річний цикл (октаетеріди), в якому місяць вставлявся в 3-му, 5-му і 8-му році. (В Афінах його введення приписують Солону в 594 році до н.е.)

- Після Метона (приблизно з 430 року до н.е.)

У 432 році до н.е. астроном Метон запропонував більш точний 19-річний цикл з 7-ю вставними місяцями, але цей цикл входив в вживання повільно і так до кінця і не прижився.

Також в давньогрецькому літочисленні використовуються Олімпіади. На основі цих 4-річних інтервалів між змаганнями в Олімпії, давньогрецькі історики використовували рахунок часу в своїх працях.

Нижче представлені назви місяців з календарів деяких грецьких міст. Місяці розташовані в порядку аттичного (Афінського) календаря. Перший місяць списку позначається I, а місяць, в який вставляється додатковий місяць - зірочкою (*). [6]

Таблиця 1.7.

Назви місяців давньогрецького календаря

Відповідно сучасному	Афіны	Делос	Мілет	Дельфи	Беотія	Македонія
Липень-Серпень	I.Hekatombaion	I.Hekatombaion	Panemos	I.Apellaios	Hippodromios	Loios
Серпень-Вересень	Metageitnion	Metageitnion	Metageitnion	Boukatios	Panamos	Gorpaions
Вересень-Жовтень	Boedromion	Bouphonion	Boedromion	Boathoos	Pamboitios	Hyperbeteitios

Жовтень- Листопад	Pyanepsion	Apatourion	Pyanepsion	Heraios	Damatrios	I.Dios
Листопад- Грудень	Maimakterion	Aresion	Apatourion	Daidophorios	Alalkomenios*	Apellaios
Грудень- Січень	Poseideon*	Poseideon	Poseideon	Poetropios*	I. Boukatios	Auduaios
Січень- Лютий	Gamelion	Lenaion	Lenaion	Amalios	Hermaios	Peritios
Лютий- Березень	Anthesterion	Hieros	Anthesterion	Bysios	Prostaterios	Dystros
Березень- Квітень	Elaphebolion	Galaxion	I. Artemision	Theoxenios	Agrionios	Xandikos
Квітень- Травень	Mounychion	Artemision	Taureon	Endyspoetropios	Thioyios	Artemios
Травень- Червень	Thargelion	Thargelion	Thargelion	Herakleios	Homoloios	Daisios
Червень- Липень	Skiophorion	Panamos*	Kalamaion	Paaios	Theilouthios	Panemos

Римський календар. Календар використовувався в Римському царстві і республіці.

Існує думка, що найдавніший римський календар цар Ромул запозичив у греків. Він містив 10 місяців, першим місяцем вважався березень. 3 з 10 місяців мали по 31 дню, інші - 30, що в підсумку давало 304 дні.

До кінця VIII століття до н.е місяці не мають назв, а позначаються порядковими номерами. Потім деякі місяці отримують свої назви: I - Martius, II - Aprilius, III - Maius, IV - Junius, інші зберігають свої числові позначення:

V - Quintilis, VI - Sextilis, VII - September, VIII - October, IX - November, X - December. Деякі назви місяців збереглися в багатьох мовах і існують понині. Березень, Травень і Квінтіліс мали по 31 дню.

Далі, на рубежі VII і VI століть до н.е. відбувається календарна реформа, яка приписується Нумі Помпілію. Внаслідок запозичення Етрурського календаря, в якому рік ділиться на 12 місяців, в Давньоримський календар додаються 2 місяці - Januarius і Februarius. Тепер рік складається з 355 днів: 4 місяці по 31 дню, 7 місяців по 29 днів і один місяць в 28 днів. Кожні кілька років додається додатковий місяць, таким чином в році може бути 355, 377 або 378 днів.

Нижче наведена таблиця місяців після реформи Нумі Помпілія:

Таблиця 1.8.

Місяці римського календаря

Назва	Кількість днів	Примітка
Martius	31	На честь бога Марса
Aprilis	29	Лат. Aperire - відкривати, початок весни
Maius	31	На честь італійської богині Майї
Junius	29	На ім'я богині Юнони
Quintilis, пізніше Julius	31	п'ятий; з 44 року до н.е. - Julius, в честь Юлія Цезаря
Sextilis, пізніше Augustus	29	шостий; з 8 року до н.е. - Augustus, в честь Октавіана Августа
September	29	Сьомий
October	31	Восьмий
November	29	Дев'ятий
December	29	Десятий

Januarius	29	На ім'я бога Януса
Februarius	28	Місяць очищень, від лат. Februare - очищати, приносити жертву спокутування в кінці року
Mercedonius; Mensis intercalaris	27 або 28	Проміжний місяць, вводився за рішенням Великого понтифіка раз в декілька років; присвячувався богині торгівлі Мерцедоні

Римляни використовували восьмиденні тижні - нундіни, в яких дні позначаються латинськими літерами: A, B, C, D, E, F, G, H. Однак з експансією Риму на Схід, вже в I столітті до н.е., нундіни замінюються на семиденні тижні.

До 291 року до н.е. (Дата появи Сонячного годинника в Римі) римляни не поділяли добу на годинник. Доба ділилася на дві половини по 12 годин.

В якості літочислення ведуться списки консулів, рік позначався по іменах двох консулів поточного року. Обрання консулів відбувається щорічно.

В епоху Августа в вживання також входить літочислення від ймовірного року заснування Риму - ad Urbe condita. [6,9]

Юліанський календар. Календар, розроблений групою олександрійських астрономів на чолі з Созігеном, був введений в Римській республіці з 1 січня 45 року до н.е. Варто відзначити, що Юлій Цезар не просто вдосконалив старий римський календар, навпаки, він відмовився від нього і замість нього встановив сонячний календар з 365.25 днів.

Причиною цієї реформи стало те, що за часів приходу Цезаря до влади, календарні дати сильно відрізнялися від очікуваних пір року. Метою реформи було відновлення синхронізації календаря з порами року і

уникнення цієї проблеми в майбутньому. Для цього, Цезар додав 90 днів до поточного року, після чого місяці і пори року були синхронізовані. Потім, з 1 січня 45 року до н.е. був введений сонячний рік з 365 днів, замість старого 355-денного. Ці десять додаткових днів були додані в кінці різних місяців так, що б дати свят не зрушилися. Вибір 1 січня, як дати нового року було обумовлено тим, що в цей день, ще з 153 року до н.е. обрані коміціями консули вступали на посаду.

Також реформа торкнулася додаткового місяцю. До реформи, місяць Мерцедоній, вставлявся в кінці лютого раз в декілька років. Причому рішення про це приймав Великий Понтифік. Тепер же, замість додаткового місяця, кожні 4 роки, після дня VI Kal. Mart. (Тобто Шостий день перед березневими календами), вставлявся додатковий день. Цей день називався *bis sextum Kal. Mart.* (Тобто Двічі шостий день перед березневими календами). Пізніше рік, який доповнювався таким днем почали називати *annus bissextus*, звідки і з'явилося високосний "високосний". Однак, після смерті Цезаря, календар не функціонував правильно, через те, що понтифіки вставляли додатковий день з інтервалом в 3 роки, а не з покладеним інтервалом в 4 роки. Цю помилку виправив наслідник Цезаря Август: з 9 року до н.е. по 8 рік н.е. додатковий дня не вставлявся. Таким чином, починаючи з 8 року н.е. календар почав нормально функціонувати.

Крім додавання 2 нових місяців, реформа змінила назви одного, а потім і ще одного місяців. Так, до цього не мав назви *Quintilis*, став називатися *Julius* (в честь Цезаря). Потім, свого часу Октавіан Август також дав своє ім'я одному з місяців: *Sextilis* став іменуватися *Augustus*.

Юліанський календар не був уніфікованим по всій Римській Імперії. Так, на заході Імперії календар вводився без змін, а ось в Східній частині, де грецька мова була офіційною, назви і тривалість місяців пристосовувалася до місцевих звичаїв. [6,9]

Нижче наведена таблиця місяців початкового юліанського календаря разом з назвами місяців деяких локальних варіацій:

Таблиця 1.9.

Місяці Юліанського календаря

Початкова назва	Антиохія	Лікія та Сідон	Тір
October	Hyperberetaios	Loos	Dios
November	Dios	Gorpaios	Apellaios
December	Apellaios	Hyperberetaios	Audynaios
Januarius	Audynaios	Dios	Peritios
Februarius	Peritios	Apellaios	Dystros
Martius	Dystros	Audynaios	Xanthikos
Aprilis	Xanthikos	Peritios	Artemisios
Maius	Artemisios	Dystros	Daisios
Junius	Daisios	Xanthikos	Panemos
Julius	Panemos	Artemisios	Loos
Augustus	Loos	Daisios	Gorpaios
September	Gorpaios	Panemos	Hyperberetaios

Сучасні календарі

У сучасному світі не існує уніфікованої календарної системи. У той час як більшість країн світу використовують Григоріанський календар, є країни, які використовують інші календарі. Основна причина цього світові релігії, які міцно пов'язані з нинішніми календарями. Так, переважно, християнські країни використовують Григоріанський календар, ісламські країни використовують Ісламський календар, а деякі буддійські країни - Буддійський.

Далі будуть наведені деякі сучасні календарі, які використовуються в сучасних країнах світу:

Григоріанський календар. Був введений папою римським Григорієм XIII 4 жовтня 1582 року замінив собою Юліанський календар.

Виділяють 2 причини Григоріанської реформи:

- Першою причиною була Пасха. Після багатовікових суперечок, було встановлено правила святкування Великодня. Для обчислення дати цього свята було необхідно знайти весняний повний Місяць, який настає відразу після весняного рівнодення. Таким чином, в календар було внесене Рухоме свято, адже Юліанський календар заснований на Сонячному році і ніяк не пов'язан з рухом Місяця і повторенням його фаз. Весняний повний Місяць з кожним роком випадав на різні числа Юліанського календаря. Однак ідея про Великдень, змушувала тримати великодній повний Місяць близько часу початку Весни. Нікейський собор у 325 році оголошує датою весняного рівнодення 21 березня, що і створює подальші проблеми. З огляду на те, що довжина календарного року не співпадала точно з довжиною сонячного року, весняне рівнодення зміщувалося на більш ранні дати. Так, приблизно кожні 125 років рівнодення зміщувалося на 1 день. І у XVI столітті фактичною датою рівнодення було 11-те березня, а не 21-ше - вказане Нікейським собором. Таким чином, свято проходило як мінімум на 10 днів раніше, ніж починалася весна, і з кожним роком наближалось до літа, втрачаючи зв'язок зі своїм початковим визначенням. Це і стало першою причиною реформи.
- Другою була не ідеальність системи обчислення Повного Місяця.

На відміну від Юліанського календаря, в Григоріанському календарі, календарний рік дорівнює 365,2425 добам. Таким чином тривалість не високосного року дорівнює 365 діб, а високосного - 366 діб.

Спочатку різниця між Григоріанським і Юліанським календарями становила 10 днів, але вона збільшується через різну кількість високосних років. На даний момент різниця становить 13 днів.

Григорій створив систему розподілу високосних років:

- Рік, номер якого кратний 400 - високосний.
- Інші роки, номери яких кратні 100 - невисокосні.
- Інші роки, номер яких кратний 4 - високосні.
- Решта року - не високосні.

Завдяки цьому похибка в добу з датою рівнодення накопичиться приблизно за 10 тисяч років (в юліанському - за 128 років).

Спочатку, календарна реформа викликала масові протести серед вчених. Держави переходили з юліанського календаря на григоріанський в різний час. Першими, хто перейшов на григоріанський календар були: Іспанія, Португалія, Італія, Річ Посполита, Франція. Останньою країною, що перейшла на григоріанський систему, на даний момент є Саудівська Аравія.

Зараз Григоріанський календар використовується в більшості країн світу. [9]

Буддійський календар. Сонячно-місячний календар, заснований на індуїстському календарі. У буддійському календарі місяці прив'язані до фаз Місяця, а обчислення року прив'язані до сонячного року.

Літочислення в буддійському календарі ведеться з 543 року до н.е. Саме цією датою вважається рік відходу Будди Гаутами в нірвану. Таким чином, буддійське літочислення випереджає григоріанське на 543 роки.

Новий рік починається в грудні, так як день Спадаючого Місяця припадає на перший день цього місяця.

Календар має два типи місяців: синодичний і сидеральний місяці. 12 синодичних місяців становлять рік. 27 сидеральних місяців використовуються

для астрологічних розрахунків. Кількість днів у місяці чергується між 29 і 30. Місяці нумеруються по порядку.

Календар має деякі відмінності, в залежності від регіону. Використовується в країнах де поширений буддизм тхеравади (Таїланд, Лаос, Камбоджа, М'янма, Шрі-Ланка). Використовувався в Тибеті, поки той не втратив свій суверенітет.

На основі буддійського календаря проводиться літочислення, а також визначаються дати буддійських свят, багато з яких є державними.

Ефіопський календар. Сонячний календар, використовується в Ефіопії. В його основі лежить Олександрійський календар. Містить 12 місяців по 30 днів, в кінці року додаються 5 або 6 епогоменальних днів. Кожен четвертий рік високосний. Початком доби вважається світанок. Новий рік починається 25 березня.

Літочислення починається з Ери Втілення (29 серпня 8 рік н.е.), таким чином, Ефіопський календар відстає від Григоріанського календаря на 8 років.

Нижче наведені назви місяців та їх назви згідно Григоріанському календарю:

Таблиця 1.10.

Місяці Ефіопського календаря

Амхарський	Початок місяця
መስከረም (Mäskäräm)	11 вересня
ጥቅምት (Teqemt)	11 жовтня
ኅዳር (Hedar)	10 листопада
ታኅሣሥ (Tahsas)	10 грудня
ጥር (T'er)	9 січня

የካቲት (Yäkatit)	8 лютого
መጋቢት (Mägabit)	10 березня
ሚያዝያ (Miyazya)	9 квітня
ግንቦት (Genbot)	9 травня
ሰኔ (Säne)	8 червня
ሐምሌ (Hamle)	8 липня
ከሐሌ (Nähase)	7 серпня
ኧተሜ (Pagumän)	6 вересня

Єврейський календар. Місячно-сонячний календар. Його коріння сягає до вавилонського календаря, що віддзеркалюється в назві місяців. Є офіційним календарем в Ізраїлі. По ньому ведуть літочислення і датування в документах, визначають дати свят. Новий рік, як і місяці, починається з Молодика. Літочислення починається з 6 (7) жовтня 3761 року до н.е.

Історія календаря має три періоди:

- Біблійний, від Мойсея до Ездри і Неємії (1500 - 450 рік до н.е.)

У цей період початок місяців і свят визначається виключно за спостереженням за фазами місяця.

- Післябіблійний (до руйнування другого Храму в 70 році до н.е.)

У цей період початок місяців визначається, частково за спостереженнями, частково за обчисленнями.

- Талмудичний, до укладення Талмуда (70-500 рік н.е.)

У цей період, після постанови Гілея Другого в 359 році, використовуються тільки математичні обчислення.

Рік (івр. שנה פשוטה, шана пшута) складається з 12 синодичних місяців, а у високосному році (שנה מעוברת, шана меуберет) додається додатковий місяць. Також в календарі є додаткова класифікація для року:

- правильний (קסדרה, ке-сидру), коли всі 12 місяців складаються поперемінно з 30 і 29 днів;
- достатній (שלמה, шолома), коли місяць Хешван, який зазвичай має 29 днів, має 30 днів;
- недостатній (חסרה, Хасер), коли місяць Кіслев, який зазвичай має 30 днів, складається з 29 днів.

Таким чином, рік може бути: правильним / достатнім / недостатнім, високосним / невисокосним. Внаслідок цього рік може мати 6 різних тривалостей:

Таблиця 1.11.

Типи року

Типи років	Днів у простому році	Днів у високосному році
Правильний	354	384
Достатній	355	385
Недостатній	353	383

Зважаючи на особливості іудаїзму, від першого року календаря - нісана, йде відлік усіх свят. Однак, Рош Ха-Шана (Новий рік) відзначається в 7-му місяці Тішрі, від нього ж ведеться літочислення.

Таблиця 1.12.

Місяці Єврейського календаря

Місяць (застар. назва, згадувана в Старому Заповіді)	Кількість днів	Зодіакальне сузір'я	Вавілонська назва
Нісан (Авів)	30	Овен	Нісану
Іяр (Зів)	29	Телець	Аяру

Сіван	30	Близнюки	Сіману
Тамуз / Таммуз	29	Рак	Ду'uzu
Ав	30	Лев	Абу
Элул	29	Діва	Улулу
Тішрей (Эйтаним)	30	Ваги	Ташріту
Хешван (Мархешван, Буль)	29 или 30	Скорпіон	Арахсамна
Кіслев (Хіслев)	29 или 30	Стрілець	Кісліму
Тевет	29	Козерог	Тебету
Шват	30	Водолій	Шабату
Адар	29*	Риби	Адару
Адар бет	29	(тільки у високосному році)	Адару

* У високосному році місяць Адар перший (Адар алеф) має 30 днів, а Адар другий (Адар бет) - 29 днів, в той час як в звичайному невисокосному році Адар алеф складається з 29 днів.

Ісламський календар. Місячний календар. Також іменується календарем Хіджри. Раніше використовувався в якості офіційного календаря Саудівської Аравії. Зараз же використовується мусульманами по всьому світу для визначення дат релігійних свят.

Літочислення в ньому ведеться від 16 липня 622 року н.е. Ця дата іменується Хіджра, в цей день пророк Мухаммед і його перші послідовники переселилися з Мекки до Медіни.

Доба починається в момент заходу сонця, початком місяця вважається неоменія - день, коли серп Місяця можна побачити у вечірні сутінки вперше після Повного Місяця.

Календар складається з 12 синодичних місяців, довжиною в 29, 5306 днів. Таким чином, в місяці може бути як 29 днів, так і 30 днів. В сумі календарний рік має 354, 3671 доби, в той час як Місячний рік містить 354 доби. Через цю помилку в 0,3671 доби, Молодик буде зрушуватися на більш пізні дати, внаслідок чого початок місяців (в місячному календарі початок кожного місяця має співпадати з Молодиком) буде зрушуватися. Щоб повернути їх на місце, слід вставляти додатковий день приблизно кожні 3 роки. Щоб знайти такий порядок чергування звичайних і високосних років, при якому початок місяців не сильно зміщувався від Молодика, було розроблено два окремих цикли:

- Турецький цикл.

У цьому циклі вставка високосів має наступну пропорцію 3/8. Тобто, на 8 Місячних років вставляється 3 високоси. Високосні роки припадають на 2, 5, 7 роки.

- Арабський цикл

Тут вставка високосів рівна 11/30. Тобто, на 30 Місячних років вставляється 11 високосів. Високосні роки припадають на 2, 5, 7, 10, 13, 16, 18, 21, 24, 26, 29 роки.

Додатковий день у високосному році додається до місяця Зуль-хіджа.

З огляду на те, що рік ісламського календаря на 10 або 11 днів менше сонячного року, місяці відносно сезонів зміщуються. Таким чином, наприклад, місяці, які припадають на зиму, незабаром будуть припадати на літо.

Нижче наведені назви місяців та їх значення:

Таблиця 1.13.

Місяці Ісламського календаря

Місяць	Значення
Мухаррам	“Заборонений”

Сафар	“Жовтий”
Рабі’у ль-авваль	В давнину означав “осінь”
Рабі’у с-сани	“Другий рабі”
Джумада аль-уля	“Замерзати”
Джумада аль-ахіра	“Останній”
Раджаб	“Боятися”
Ша’бан	“Розділятися”
Рамадан	“Бути розпеченим”
Шавваль	“Зніматися з місця”
Зуль-ка’да	“Сидіти, перебувати на місці”
Зуль-хіджа	“Здійснювати паломництво”

Місяці зуль-када, зуль-хіджа, мухаррам і раджаб в доісламські часи вважалися забороненими місяцями (араб. الحرم الأشهر), так як в ці місяці не можна було вести війни, вбивати і полювати навколо Кааби. В ісламі ж ці місяці вважаються священними. [9]

Іранський календар. Сонячний календар, розроблений астрономами під керівництвом Омара Хайяма. Як і Мусульманський календар, веде літочислення від Хіджри, проте ґрунтується на сонячному році. Початком року вважається день весняного рівнодення (Навруз). Календар є офіційним у Ірані і Афганістані.

Іранський календар корінням сходить до більш давнішого іранського календаря. Він складався з 6 сезонів, кожен з яких відповідав двом синодичним місяцям. Потім, зіткнувшись з месопотамською культурою,

перси синхронізували свій календар з вавілонським календарем. Тепер рік починався з весняного рівнодення, а календар складався з 12 місяців, раз в 6 років вставлявся додатковий місяць.

Потім, в V столітті до н.е. Ахеменідська адміністрація вводить новий календар - Зороастрійський. Цей календар, на відміну від минулого, заснований на сонячному році і зроблений за єгипетським зразком, складається з 12 місяців по 30 днів. Назви місяців даються на честь шанованих зороастрійських язатів. Також до календарного року додаються 5 додаткових днів. Цей календар і стає основою сучасного Іранського календарю, а назви його місяців зберігаються, декілька видозміненими досі.

Рік ділиться на чотири сезони по три місяці кожен:

- **Весна** (перс. بهار [bæhr̩], пушту پسرلی [psar'lai]): фарвардін, ордібехешт, Хордад
- **Літо** (перс. تابستان [tobestɒn], пушту دوبي ['dobai]): тір, мордад, Шахривар
- **Осінь** (перс. پایز [pəjiz], пушту منی ['məni]): Мехр, Абан, азар
- **Зима** (перс. زمستان [zemestɒn], пушту ژمی ['zəmai]): дей, Бахман, есфанд

Тиждень починається в суботу і закінчується в п'ятницю:

- Субота - **Шанбе** (перс. شنبه);
- Неділя - **Йекьшанбе** (перс. یکشنبه);
- Понеділок - **Дошанбе** (перс. دوشنبه);
- Вівторок - **Сешанбе** (перс. سه شنبه);
- Середа - **Чахаршанбе** (перс. چهارشنبه);
- Четвер - **Панджшанбе** (перс. پنجشنبه);
- П'ятниця - **Джом'е** (перс. جمعه) або Одіне (перс. آدینه)

Нижче наведені назви місяців і їх значення:

Таблиця 1.14.

Місяці Іранського календаря

Назва в Зороастрійському календарі	Сучасна назва	Значення
Frawašīnaṃ	Farvardīn	Фраваші
Ašahe Vahištahe	Ordībehešt	Аши Вахішта
Haurvatātō	Xordād	Хаурватат
Tištryehe	Tīr	Тіштрія
Amərətātātō	Mordād	Амеретат
Xšaθrahe Wairyehe	Šahrīvar	Хшатра Ваірья
Miθrahe	Mehr	Мітра
Araṃ	Ābān	Води
Āθrō	Āzar	Огонь
Daθušō	Dey	Творець
Waṇhēuš Manaṇhō	Bahman	Воху Мана
Spəntayā Ārmatōiš	Esfand	Спента Армаіті

Календар чучхе. Збігається з календарем Міньго. Використовується в КНДР з 8 липня 1997 року, поряд з літочисленням від Різдва Христового. За перший рік календаря береться рік народження Кім Ір Сена. При написанні дат використовуються обидва літочислення в спільній формі.

Китайський календар.

Складається з двох окремих календарів:

сонячного (сільськогосподарського) 农历 і **сонячно-місячного** 阳历.

Використовується в Китаї для визначення дат традиційних свят і для визначення початку сільськогосподарських робіт. Більшість східних календарів засновані на китайському і мають лише невеликі відмінності.

Спочатку сонячний рік в стародавньому Китаї ділився на 4 пори року, однак в спробі узгодити синодичні місяці з сонячними роками, рік розділили на 12 сонячних періодів. Цей сонячний календар визначав сільськогосподарський сезон за положенням Сонця на екліптиці.

Нижче наведен сонячний календар:

Таблиця 1.15.

Китайський сонячний календар

Екліптична довгота	Переклад	Китайська назва	Приблизна дата в Григоріанському календарі
315	Початок весни	立春 лічунь	4-5 лютого
330	Дощова вода	雨水 юйшуй	19—20 лютого
345	Пора пробудження личинок	驚蟄 (惊蛰) цзінчжэ	5—6 березня
0	Весняне рівнодення	春分 чуньфэнь	20—21 березня
15	Ясно і світло	清明 цінмін	4—5 квітня
30	Хлібні дощі	穀雨 (谷雨) гуюй	20—21 квітня
45	Початок літа	立夏	5—6 травня

		лися	
60	Малий достаток	小滿 (小满) сяомань	21—22 травня
75	Колосіння хлібів	芒種 (芒种) манчжун	5—6 червня
90	Літнє сонцестояння	夏至 сячжи	21—22 червня
105	Мала спека	小暑 сяошу	7—8 липня
120	Велика спека	大暑 дашу	23—24 липня
135	Початок осені	立秋 лицю	7—8 серпня
150	Припинення спеки	處暑 (处暑) чушу	23—24 серпня
165	Білі роси	白露 байлу	7—8 вересня
180	Осіньне рівнодення	秋分 цюфэнь	23—24 вересня
195	Холодні роси	寒露 ханьлу	8—9 жовтня
210	Випадання інею	霜降 шуанцзян	23—24 жовтня
225	Початок зими	立冬 лідун	7—8 листопада
240	Малі сніги	小雪	22—23 листопада

		сяосяюэ	
255	Великі сніги	大雪 дасюэ	7—8 грудня
270	Зимове сонцестояння	冬至 дунчжи	21—22 грудня
285	Малі холоди	小寒 сяохань	5—6 січня
300	Великі холоди	大寒 дахань	20—21 січня

У місячно-сонячному календарі, Молодик, який вважається початком місяців, припадає на певні пари сільськогосподарських сезонів. Якщо на пару сільськогосподарських сезонів випадає два Молодика, то один і той же місяць рахується двічі.

Нижче наведена таблиця місяців за Місячно-сонячним календарем:

Таблиця 1.16.

Китайський місячно-сонячний календар

Номер місяця	Початок місяця	Фенологічна назва	Назва по земній гілці	Сучасна назва
1	21 січня — 20 лютого	кит. 陬月 (zōuyuè) — кутовий місяць	кит. 寅月 (yínyuè) — місяць тигра	кит. 正月 (zhēngyuè) — перший місяць
2	20 лютого — 21 березня	кит. 杏月 (xìngyuè) — місяць абрикосу	кит. 卯月 (mǎoyuè) — місяць кролика	кит. 二月 (èryuè) — другий місяць
3	21 березня	кит. 桃月	кит. 辰月	кит. 三月

	— 20 квітня	(táoyuè) — місяць персику	(chényuè) — місяць дракона	(sānyuè) — третій місяць
4	20 квітня — 21 травня	кит. 梅月 (méiyuè) — місяць квітки сливи	кит. 巳月 (sìyuè) — місяць змії	кит. 四月 (sìyuè) — четвертий місяць
5	21 травня — 21 червня	кит. 榴月 (liúyuè) — гранатовий місяць	кит. 午月 (wǔyuè) — місяць коня	кит. 五月 (wǔyuè) — п'ятий місяць
6	21 червня — 23 липня	кит. 荷月 (héyuè) — місяць лотоса	кит. 未月 (wèiyuè) — місяць кози	кит. 六月 (liùyuè) — шостий місяць
7	23 липня — 23 серпня	кит. 蘭月, 兰月 (lányuè) — місяць орхідеї	кит. 申月 (shēnyuè) — місяць мавпи	кит. 七月 (qīyuè) — сьомий місяць
8	23 серпня — 23 вересня	кит. 桂月 (guìyuè) — місяць османтусу	кит. 酉月 (yǒuyuè) — місяць півня	кит. 八月 (bāyuè) — восьмий місяць
9	23 вересня — 23 жовтня	кит. 菊月 (júyuè) — місяць хризантеми	кит. 戌月 (xūyuè) — місяць собаки	кит. 九月 (jiǔyuè) — дев'ятий місяць
10	23 жовтня — 22 листопада	кит. 露月 (lùyuè) —	кит. 亥月 (hàiyuè) — місяць свині	кит. 十月 (shíyuè) —

		місяць роси		десятий місяць
11	22 листопада — 22 грудня	кит. 冬月 (dōngyuè) — місяць зими; кит. 葭月 (jiāyuè) — місяць очерету	кит. 子月 (zǐyuè) — місяць щура	кит. 十一月 (shíyīyuè) — одинадцятий місяць
12	22 грудня— 21 січня	кит. 冰月 (bīngyuè) — місяць льоду	кит. 丑月 (chǒuyuè) — місяць бика	кит. 腊月, 臘 月 (làyuè) — останній місяць

Японський календар. Місячно-сонячний календар, в його основі лежить Китайський місячний календар. До 1873 року був офіційним календарем Японії. Однак з 1 січня 1873 року Японія перейшла на Григоріанський календар. Зараз традиційний календар використовується в церемоніальних і культурних цілях, а традиційне літочислення по імператорам використовується японцями в повсякденному житті.

Що стосується літочислення, за фактом на даний момент, в Японії існують три системи літочислення:

- від заснування Японії (яп. 皇紀 до: ки) - стародавня календарна система, точкою відліку є 660 рік до н. е., коли, згідно з легендою, імператор Дзімму заснував Японську державу;

- від початку правління імператора (яп. 元号 Генг:) - роки обчислюються від року початку правління імператора, наприклад, 2012 рік є 24 роком Хейсей;
- від Різдва Христового (Наша ера) (яп. 西暦 сейрекі).

Однак зараз використовуються лише дві останні, так як перша система використовувалася лише з 1873 року і до кінця Другої світової війни.

Кожен місяць японського календаря має власні, лише йому притаманні назви, що відображають прикмети часу року, характер сільськогосподарських робіт, звичаї і обряди, характерні для цього місяця.

Нижче наведені місяці Японського традиційного календарю:

Таблиця 1.17.

Японський традиційний календар

Місяць	Японський	Кирилиця	Значення
1-й місяць	睦月	<i>муцукі</i>	місяць дружби
2-й місяць	如月 або 衣更着	<i>кісарагі</i> або <i>кінусарагі</i>	місяць одягу, або <i>тюсюн</i> - середина весни
3-й місяць	弥生	<i>яйьоі</i>	місяць произрастання
4-й місяць	卯月	<i>удзукі</i>	місяць унохана або уцугі, унохана - це чагарник, різновид дейції
5-й місяць	皐月 або 早月 або 五月	<i>сацукі</i>	місяць рисових посівів, або <i>тюка</i> - середина літа

6-й місяць	水無月	<i>мінацукі</i> або <i>мінадзукі</i>	безводний місяць
7-й місяць	文月	<i>фумідзукі</i> або <i>фудзукі</i>	місяць написання поезії
8-й місяць	葉月	<i>хадзукі</i>	місяць (опадаючого) листя, або цукімідзукі - місяць милування Місяцем, або <i>тюсю</i> - середина осені
9-й місяць	長月	<i>нагацукі</i>	місяць довгих ночей, або <i>кікудзукі</i> - місяць хризантем
10-й місяць	神無月	<i>камінадзукі</i> або <i>каннадзукі</i>	місяць без богів, або <i>камінарі</i> - місяць без грому, або <i>камінас</i> - місяць приготування sake
11-й місяць	霜月	<i>сімоцукі</i>	місяць інею, або <i>тjото</i> - середина зими
12-й місяць	師走	<i>сівасу</i>	місяць закінчення справ

Мова Python

Історія створення

Першим, хто почав спроби створення Python в 1990 року стало Гвідо Ван Россум. Працюючи над розробкою мови ABC, він зрозумів, що хотів би створити щось нове. Перший робочий прототип Python був створений на

домашньому Макінтош Гвідо за пару вихідних. В основі нового інтерпретатора лежали деякі ідеї, запозичені у ABC. Цей прототип поширювався через інтернет. Мова отримала назву не в честь виду змій, а була названа на честь Монті Пайтона, так як Гвідо любив дивитися комедійне шоу "Повітряний цирк Монті Пайтона".

У 1996 році до розробки підключився Стів Маєвський, який на той момент вів свій блог "Порівняльна критика мов програмування". Через великий потенціал Python і вільну модель поширення через Інтернет, мова знайшла ядро послідовників - людей, зацікавлених в її розвитку. На той момент Python мав вигляд невеликого інтерпретатора з малою кількістю функцій і відсутністю ООП, що нікого не влаштовувало, і що послужило відмінною мотивацією для подальшого розвитку.

Через деякий час, Гвідо пропонують посаду в корпорації CNRI, що знаходиться в Америці. Гвідо погоджується і залишає Голландію. Працюючи над проектами компанії, він частково використовує Python для вирішення завдань і у вільний час займається його розвитком.

Так, Python розвивався до 1999 року, отримавши версію 1.5.2. Однак компанія все більше завантажувала його роботою, що заважало йому займатися розвитком свого дітища. Це спонукало його шукати спонсора, який дасть можливість працювати тільки над розвитком мови. Таким спонсором стала фірма BeOpen. Працюючи з BeOpen, він випускає версію 2.0, багато хто стверджує, що випуск цієї версії дав сильний поштовх у розвитку спільноти, так як процес розвитку мови став більш відкритим. Так як спільнота вимагала можливості участі в розробці коду для Python, Гвідо перевів всі дані на SourceForge.

Незабаром у Гвідо почалися проблеми з BeOpen, так як ті почали вимагати працювати його старанніше і приносити їм гроші. Гвідо йде з компанії, що на його думку стало переломним моментом його життя.

Наступним спонсором ставала компанія Digital Creations, так як це була п'ята пропозиція від них, Гвідо погоджується. Команда розробників Python отримала великі можливості, що відразу дало результат - випускається версія 2.1. В Python з'являються нові об'єкти з мовою closures і ієрархія: функції могут бути вкладені одна в одну, зберігаючи при цьому доступ до змінних оточуючих функцій. В майбутньому це сильно змінить мову і поліпшить її підходи до способу програмування. [10]

Особливості Python

Основні властивості.

Python динамічно типізована, регістро залежна, об'єктивно-орієнтована мова.

Синтаксис.

В Python немає операторних дужок, для виділення блоків використовуються відступи, а вхід в блок операторів здійснюється двокрапкою. Для створення однострочного коментаря використовується знак фунта «#», а многострочного - послідовністю з трьох подвійних лапок «""""». Щоб присвоїти значення змінній використовуються знак «=», для порівняння – «==».

Структури даних.

Python містить такі структури даних як списки (lists), кортежі (tuples) і словники (dictionaries). Списки - одномірні масиви (для створення багатовимірного списку потрібно створити список включаючий інші списки), кортежі - незмінні списки, словники - теж списки, але індекси можуть бути будь-якого типу, а не тільки числовими. Списки в Python можуть містити

дані будь-якого типу. Масиви починаються з індексу 0, а останній елемент можна отримати за індексом -1. Змінним можна присвоювати функції.

Для використання частини масиву слід задати перший і останній індекс через двокрапку «:». Так, ви отримаєте частину масиву, від першого індексу до іншого не включно. Якщо не вказати перший індекс, то відлік почнеться з початку масиву, а якщо не вказати другий - масив буде злічений до останнього елементу.

Рядки.

Рядки в Python виділяються лапками. У шаблон рядка можна підставити елементи кортежу або словника. Знак відсотка «%» між рядком і кортежем, замінює в рядку символи «% s» на елемент кортежу. Словники дозволяють вставляти в рядок елемент під заданим індексом. Для цього треба використовувати в рядку конструкцію «% (індекс) s». У цьому випадку замість «% (індекс) s» буде підставлено значення словника під заданим індексом.

Оператори.

Оператори while, if, for складають оператори переміщення. У Python немає аналога оператору select. В операторі for відбувається порівняння змінної і списку. Щоб отримати список цифр до числа <number> - використовуйте функцію range (<number>). Ось приклад використання операторів:

Функції.

Для оголошення функції служить ключове слово «def». Аргументи функції задаються в дужках після назви функції. Можна задавати необов'язкові аргументи, привласнюючи їм значення за замовчуванням. Функції можуть повертати кортежі, в такому випадку треба писати повернені значення через кому. Ключове слово «lambda» служить для оголошення елементарних функцій.

Класи.

Мова Python обмежена в множинному спадкуванні в класах. Внутрішні змінні і внутрішні методи класів починаються з двох знаків нижнього підкреслення «`__`» (наприклад «`__myprivatevar`»). Ми можемо також привласнити значення змінної класу ззовні.

Винятки.

Винятки в Python мають структуру `try-except [exceptionname]`

Імпорт.

Зовнішні бібліотеки можна підключити процедурою «`import [libname]`», де `[libname]` - назва підключається бібліотеки. Ви так само можете використовувати команду «`from [libname] import [funcname]`», щоб ви могли використовувати функцію `[funcname]` з бібліотеки `[libname]`.

Особливості.

- Умови можуть комбінуватися. `1 <a <3` виконується тоді, коли `a` більше 1, але менше 3.
- Використовуйте операцію «`del`» щоб очищати змінні або елементи масиву.
- Python пропонує великі можливості для роботи зі списками. Ви можете використовувати оператори оголошенні структури списку. Оператор `for` дозволяє задавати елементи списку в певній послідовності, а `if` - дозволяє вибирати елементи за умовою.
- Глобальні змінні оголошуються поза функцій і можуть бути прочитання без будь-яких оголошень. Але якщо вам необхідно змінити значення глобальної змінної з функції, то вам необхідно оголосити її на початку функції ключовим словом «`global`», якщо ви цього не зробите, то Python оголосить змінну, доступну тільки для цієї функції.

Переваги Python.

- Висока швидкість виконання програм.
- Можливість писати модулі для Python на C або C ++
- У стандартних бібліотеках Python є засоби для вирішення багатьох завдань.
- Кросплатформеність.
- Python підходить для розробки в різних областях програмування.
- Велике і дружнє співтовариство. [11]

Фреймворк Django

Історія створення

В кінці 2003 коли програмісти газети «Lawrence Journal-World», Едріан Холоваті (Adrian Holovaty) і Симон Віллісон (Simon Willison), почали використовувати мову Python для розробки своїх додатків. Для сайтів, журналісти і менеджери вимагали, що б нові можливості і додатки розроблялися максимально швидко. Для цього, Симон і Едріан, створюють середовище розробки, за допомогою якої вони могли створювати додатки в жорсткі терміни.

Влітку 2005 року вони випускають цю середу в вигляді ПЗ з відкритим кодом. Воно отримує ім'я Django в честь джазового гітариста Джанга Рейнхарда.

Тепер, Django це грамотний проект з відкритим вихідним кодом, над яким працюють розробники по всьому світу і яким користуються десятки тисяч користувачів. Два творця, Симон і Едріан, до сих пір керують розвитком середовища розробки, але тепер воно в більшій мірі залежить від спільний зусиль команди розробників. [12]

Архітектура

Фреймворк Django написаний на мові програмування Python, тому його структура відповідає особливостям мови. Творці реалізували в Django патерн MVC, і він застосовується в поточній версії фреймворка.

Архітектура MVC дозволяє розробнику працювати з візуальним представленням і бізнес-логікою додатка окремо. До речі, при роботі з Django фахівці частіше використовують термін MVT - Model-View-Template або модель-уявлення-шаблон. Так як вважається, що компонент view, крім своєї ролі, виконує також роль controller. Компоненти MVT можна використовувати незалежно один від одного.

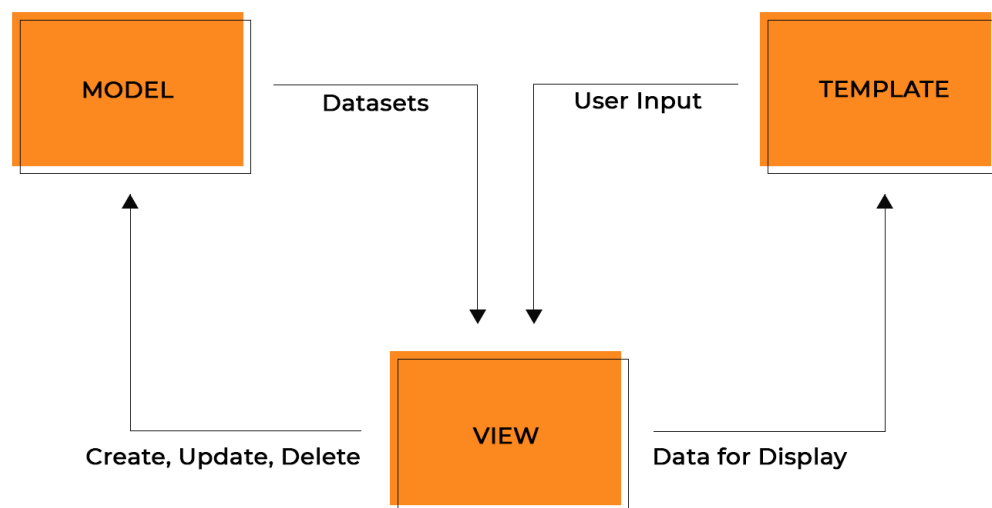


Рис. 1.1. Схема архітектури MVT в Django

Документація Django визначає модель (model) як «джерело інформації про дані, в яких містяться ключові поля і поведінка даних». Зазвичай одна модель вказує на одну таблицю в базі даних. Django підтримує бази даних PostgreSQL, MySQL, SQLite і Oracle.

Моделі містять інформацію про дані. Ці дані представлені атрибутами або полями. Оскільки модель являє собою простий клас, вона нічого не знає про інші рівні Django. Взаємодія між рівнями відбувається через API.

Модель відповідає за бізнес-логіку, методи, властивості і інші елементи, пов'язані з маніпуляцією даними. Також моделі дозволяють розробникам створювати, читати, оновлювати та видаляти об'єкти в базі даних.

Представлення (view) вирішує три завдання: приймає HTTP-запити, реалізує бізнес-логіку, визначену методами і властивостями, відправляє HTTP-відповідь на запити. Тобто view отримує дані від моделі і надає шаблонам (templates) доступ до цих даних або попередньо обробляє дані і потім надає до них доступ шаблонам.

В Django реалізований потужний движок шаблонів і власна мова розмітки. Шаблони являють собою файли з HTML-кодом, за допомогою якого відображаються дані. Вміст файлів може бути статичним або динамічним. Шаблони не містять бізнес-логіки. Тому вони тільки відображають дані.

Переваги і недоліки

Недосвідчені фахівці вважають Django однією з багатьох систем управління контентом (CMS). Насправді це програмний інструмент, за допомогою якого можна створювати і запускати веб-додатки.

Фреймворк Django справляється з великою кількістю завдань і підвищеними навантаженнями.

Переваги

Якщо ви запитаете у кількох розробників, чому вони вибрали Django, відповіді будуть переважно однаковими. Нижче описані основні переваги фреймворка, завдяки яким він став популярним.

Розвинена екосистема

Досвідчені розробники рекомендують сприймати Django як систему. Це означає, що фреймворк зазвичай використовується з великою кількістю сторонніх додатків. Їх можна вибирати в залежності від потреб конкретного проекту.

Зрілість

Django був представлений в 2005 році. За 14 років існування він сильно змінився і удосконалився. У фреймворку постійно з'являються нові можливості, а старі удосконалюються.

Адміністративна панель

Адміністративна панель Django автоматично генерується при створенні програми. Це позбавляє розробника від необхідності створювати адмінку вручну.

За допомогою сторонніх додатків стандартну консоль управління Django можна вдосконалити і адаптувати під потреби свого проекту. Крім того, фреймворк дозволяє налаштовувати інтерфейс стандартної адміністративної панелі.

SEO-дружність

Написаний на Python код виходить читабельним і зрозумілим навіть непідготовленим людям. Це один з факторів, завдяки яким веб-додатки на Python вважаються SEO-дружніми. Django генерує семантичні URL. Їх також називають людино-зрозумілими URL або ЧПУ. У додатках на Django легко реалізуються інші функції, необхідні для пошукової оптимізації.

Розширюваність

Функціональність Django розширюється за допомогою плагінів. Це програмні модулі, які дозволяють швидко додати на сайт потрібну функцію. В офіційному каталозі є сотні плагінів, які дозволяють легко реалізувати на сайті sitemap.xml, управляти доступами, підключити платіжну систему Stripe

і так далі. При необхідності ви можете відключати або замінювати плагіни, щоб пристосувати її до поточних потреб проекту.

Бібліотеки

У популярних мовах програмування є бібліотеки, за допомогою яких зручно вирішувати спеціальні завдання. В бібліотеках можна знайти готові рішення: функції, класи, конфігурації і так далі. Завдяки таким рішенням розширюються можливості мови, а також спрощується створення додатків.

Django підтримує використання бібліотек при розробці веб-додатків. У число популярних бібліотек входять:

- Django REST Framework, який спрощує роботу з API.
- Django CMS - зручний інструмент для управління контентом.
- Django-allauth - з його допомогою реалізуються функції реєстрації, авторизації, управління обліковими записами. *ORM*

В Django реалізовано об'єктно-реляційне відображення (ORM), яке забезпечує взаємодію додатку з БД. ORM автоматично передає дані з БД, наприклад, PostgreSQL або MySQL, в об'єкти, які використовуються в коді програми.

ORM прискорює розробку прототипів і готових веб-додатків на Django. Розробнику навіть не потрібно знати мову, яка використовується для взаємодії з базами даних.

Також ORM дозволяє швидко перемикатися між базами даних з мінімальними змінами коду. Наприклад, ви можете використовувати SQLite на локальному сервері, а потім переключитися на MySQL на production-сервері. Однак для мінімізації помилок краще використовувати одну базу даних під час розробки і в продакшені. [13]

Django використовує динамічну генерацію зверстаних сторінок, використовуючи для цього власний механізм шаблонів. Для створення такої

сторінки використовується МШ Django, основними елементами якої є змінні та теги.

Синтаксис змінної: {{назва змінної}}. Під час генерації сторінки змінна замінюється результатом її обчислення.

Синтаксис тегу: {%%}. Теги відповідають за логіку в процесі генерації сторінки. Тег може реалізовувати як логіку керування або циклу, так і захватувати дані з БД або керувати доступом до інших тегів.

Недоліки

- *Не підходить для невеликих проектів*
- *Немає підтримки WebSocket за замовчуванням*
- *Монолітність*
- *Непередбачуваність поведінки деяких компонентів*

Розділ 2. Проектні рішення. Розробка додатку за допомогою фреймворку Django.

Технічне завдання на створення web-додатку

Найменування і область застосування

Результати даної роботи можуть бути використані рядовими користувачами для відстеження поточної дати, моніторингу важливих подій, а також для планування.

Підстави для розробки

Завдання на виконання дипломного проекту. Наказ по університету №317-с від 07.06.2021р.

Призначення розробки

Система призначена для підвищення ефективності планування користувачем будь-яких задач. Для надання користувачу актуальних дат, поточних задач на день/неділю/місяць і для планування нових задач.

Вимоги до функціональних характеристик

Розроблюваний web-додаток повинен забезпечувати:

- Відображення актуальних дат.
- Можливість перегляду запланованих задач.
- Можливість планування нових задач.
- Мати зручний інтерфейс.

Вимоги до надійності

Вимоги до надійності не висуваються.

Умови експлуатації

Умови експлуатації повинні відповідати встановленим законодавством України нормам для користувачів відео дисплейних терміналів.

Вимоги до складу і параметрів технічних засобів

Виходячи з технічних характеристик та умов безпеки життєдіяльності, вимоги до складу і параметрів технічних засобів повинні бути наступними:

- Монітор з розширенням не менше 1280x720 dpi;
- Процесор з тактовою частотою не менше 2ГГц та вище;
- ОЗУ не менше 2 Гб;
- Не менше 1Гб на жорсткому диску;
- Клавіатура;
- Маніпулятор типу «Мишка»;

Вимоги до інформаційної та програмної сумісності

У зв'язку з вибором фреймворку Django для інформаційної та програмної сумісності необхідна наявність наступного програмного забезпечення:

- Операційна система Windows/Linux;
- Python 3.4 и выше;
- SQLite 3.1 и выше;
- Встановлений фреймворк Django версії 3.2.4 або вище;
- Встановлені бібліотеки python datetime, python calendar;

Вимоги до маркування та упаковки

Вимоги до маркування та упаковки не висуваються.

Вимоги до транспортування та зберігання

Вимоги до транспортування та зберігання не висуваються.

Вимоги до програмної документації

Програмна документації повинна мати наступні документи:

- «Технічне завдання»
- «Пояснювальна записка»
- «Опис програми»
- «Текст програми»

Стадії та етапи розробки

Таблиця нижче містить стадії та етапи розробки web-додатку:

Таблиця 2.1.

Етапи розробки

Етапи розробки	Зміст робіт
Проектування	На цьому етапі прогнозується або задається поведінка користувача. На основі цієї поведінки розроблюється дизайн сторінок додатку, який дозволяє утримати користувача.
Розробка дизайну	На цьому етапі оформлюються графічні елементи додатку.
Верстка	На цьому етапі інтегрується текстовий зміст, графіка та програмні компоненти у єдине ціле.
Програмування	На цьому етапі розроблюються та підключаються програмні компоненти додатку, які надають користувачу необхідний функціонал.
Тестування	На цьому етапі проводиться тестування отриманого продукту
Публікація	На цьому етапі надається публічний доступ до продукту

Порядок контролю та приймання

Приймання здійснюється комісією, у яку входять – завідувач кафедри, системний адміністратор та керівник дипломної роботи.

При прийманні перевіряється працездатність програмного продукту.

Архітектура додатку

Загальна структура проекту

Розглянемо загальну структуру проекту, а також файли, які за замовчення створюються при створенні нового проекту на базі Django.

Структура проекту:

DjangoProject/

env/

Project/

manage.py

db.sqlite3

Project/

apps/

static/

templates/

__init.py__

asgi.py

settings.py

urls.py

wsgi.py

env – папка містить віртуальне оточення Python, у якому встановлений фреймворк Django.

Project – контейнер проекту.

manage.py – скрипт, який дозволяє взаємодіяти та керувати проектом Django.

Наприклад запускати локальний сервер Django.

db.sqlite3 – база даних Sqlite, містить моделі проекту. Файл бази даних створюється не одразу, а у процесі створення проекту. Django ORM користується моделями даних для взаємодії з базами даних. Після створення будь якої нової моделі, для неї потрібно створити нову міграцію – опис змін, які вносяться до бази даних. Для створення міграції використовується команда `makemigrations`, а для синхронізації міграції з базою даних – команда `migrate`. Таким чином файл бази даних, створюється під час ініціалізації першої міграції.

__init.py__ - пустий файл, який вказує Python, що поточний каталог є пакетом Python.

asgi.py – Точки входу ASGI-сумісних веб-серверів.

settings.py – файл конфігурації проекту.

urls.py – конфігурація URL.

wsgi.py – Точки входу WSGI-сумісних веб-серверів.

Призначення та складова папок *apps*, *static*, *templates* буде розглянуто далі.

Анатомія папки **apps**

Папка `apps` містить додатки, які використовуються проектом. У даному проекті є тільки один додаток, але великі проекти можуть мати декілька використовуваних додатків, які можуть використовуватися декількома проектами одразу. Тому групування таких додатків в одній папці є зручним з точки зору доступу.

Структура папки `apps`:

/apps/

Calendar/

migrations/

__init__.py

admin.py

apps.py

forms.py

models.py

tests.py

urls.py

utils.py

views.py

migrations – містить файли міграцій.

__init.py – вказує, що поточний каталог є пакетом Python.

admin.py – містить зареєстровані моделі даних для взаємодії на панелі адміністратора.

apps.py – конфігурація додатку.

forms.py – містить форми, які використовує додаток

models.py – містить моделі даних.

tests.py – використовується для тестів.

urls.py – папка локальних url додатку. Підключається до кореневого urls.py.

utils.py – містить клас Calendar, який використовується для створення календаря.

views.py – містить представлення. Представлення оброблюють запити, взаємодіють з моделями та використовують шаблони для генерації сторінок.

Анатомія папки **static**

Для роботи веб-додатку зазвичай потрібні додаткові файли(зображення, CSS-файли, Скрипти JavaScript та інші). У нашому випадку ця папка містить стилі, які використовуються для оформлення сторінок додатку.

Структура папки **static**:

/static/

assets/

admin/

css/

style.css

details.css

admin – містить статичні файли, використовувані панеллю адміністратора

css – містить файли css, використовувані сторінками додатку

style.css – базова таблиця стилів, використовується на всіх сторінках.

details.css – додаткова таблиця стилів, використовується на сторінці деталей.

Анатомія папки **templates**

Django дозволяє динамічно генерувати HTML-сторінки. Для цього використовуються шаблони. Вони містять статичний HTML та динамічні дані, рендеринг яких описується спеціальним синтаксисом. Ця папка має шаблони, які використовує веб-додаток.

Структура папки **templates**:

/templates/

base.html

Calendar/

calendar.html

details.html

events.html

base.html – базовий шаблон. Використовується для розширення інших шаблонів за допомогою конструкцій мови шаблонів Django.

Calendar – містить шаблони, які використовуються додатком Calendar.

calendar.html – шаблон календаря.

details.html – шаблон сторінки деталей.

events.html – шаблон для оброблення форм.

Структура бази даних

За замовчення Django у якості бази даних використовуєе SQLite. Вона зручна у використанні та не потребує запущеного сервера. Але за необхідності Django може використовувати більшість популярних СУБД.

База даних проекту має наступні таблиці:

Таблиця 2.2.

Список таблиць БД SQLite

Назва таблиці	Опис
Calendar_event	Таблиця, яка відповідає моделі Event
auth_group	Містить авторизовані групи
auth_group_permissions	Містить список привілеїв для груп
auth_permission	Містить список привілеїв
auth_user	Містить авторизованих користувачів
auth_user_groups	Містить групи користувачів
auth_user_user_permissions	Містить привілеї для користувача
django_admin_log	Таблиця зберігає список дій у панелі

	адміністратора
django_content_type	Містить всі моделі проекту
django_migrations	Зберігає міграції
django_sessions	Містить історію сесій
sqlite_sequence	Внутрішня таблиця, необхідна для реалізації AUTOINCREMENT. Кожній таблиці користувача яка використовує автоінкрементацію, відповідає рядок таблиці

Розглянемо детальніше таблицю Calendar_event. Вона має наступну структуру:



Field Name	Field Type
id	integer
title	varchar(200)
description	text
start_time	datetime
end_time	datetime
status	bool

Рис. 2.1. Графічне представлення таблиці Calendar_event

Опис полів:

- Id – первинний ключ;
- Title – назва події;
- Description – опис події;
- Start_time – дата початку події;
- End_time – дата закінчення події;
- Status – поточний стан події;

Установка ПЗ, необхідного для запуску додатку

Установка Python

Завантажуємо Python з офіційного сайту: <https://www.python.org/downloads>

Перед установкою вмикаємо пункт, який додає Python до системних змінних.

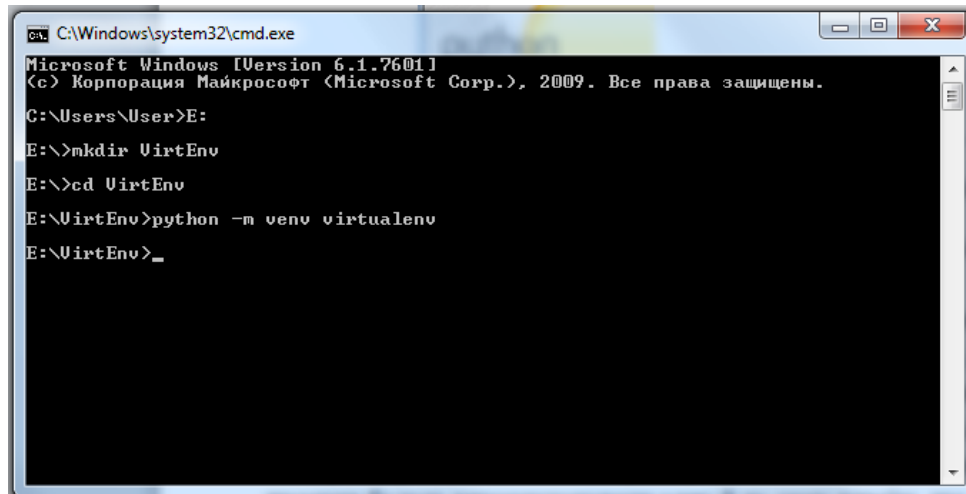


Рис.2.2. Вмикаємо пункт «Add Python 3.8 to PATH»

Далі ніяких особливих налаштувань не потребується. Після установки Python створюємо віртуальне середовище.

Віртуальне середовище Python – це спосіб ізолювати пакети для певного проекту. Воно потрібно, для запобігнення конфлікту версій. Наприклад два проекти, назвемо їх проект А та проект Б, мають залежність від однієї й тієї ж бібліотеки. Проект А запитує версію бібліотеки 1.0, у той час як проект Б використовує версію 2.0. Тут и починаються проблеми. Адже Python, не розрізняє версії у каталозі «site-packages», тому обидві версії будуть знаходитися в одному каталозі. Таким чином обидва проекти будуть використовувати одну й ту саму версію, що не є приємним.

Для створення використаємо модуль `venv`, який йде в поставці Python 3. Для створення віртуального середовища використовується команда «`python -m venv`» та назва директорії, у якій буде створено середовище.



```

C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
(C) Корпорация Майкрософт (Microsoft Corp.), 2009. Все права защищены.
C:\Users\User>E:
E:\>mkdir VirtEnv
E:\>cd VirtEnv
E:\VirtEnv>python -m venv virtualenv
E:\VirtEnv>_

```

Рис. 2.3. Створюємо віртуальне середовище

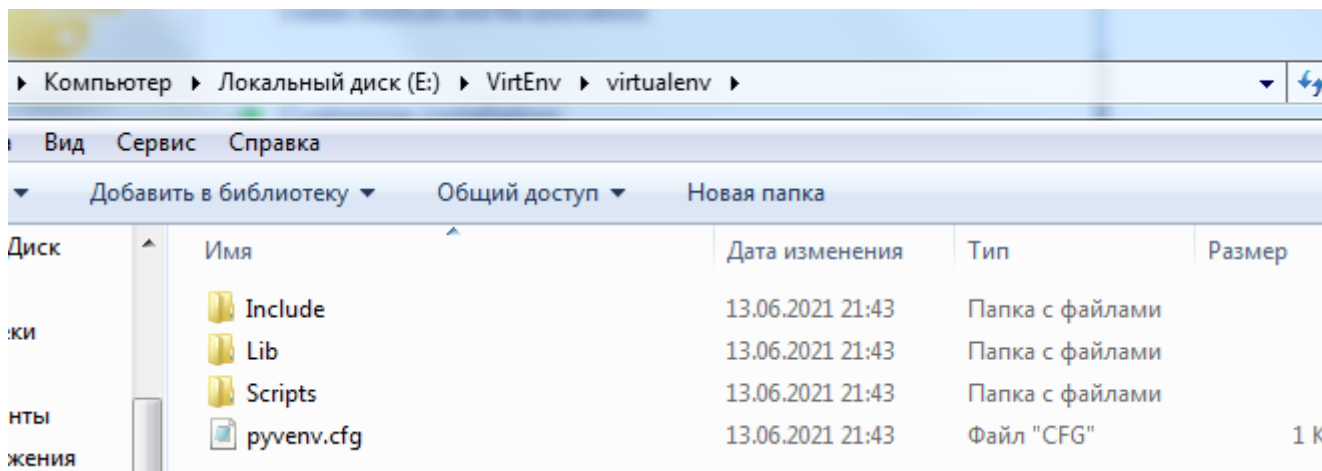


Рис. 2.4. Зміст віртуального середовища

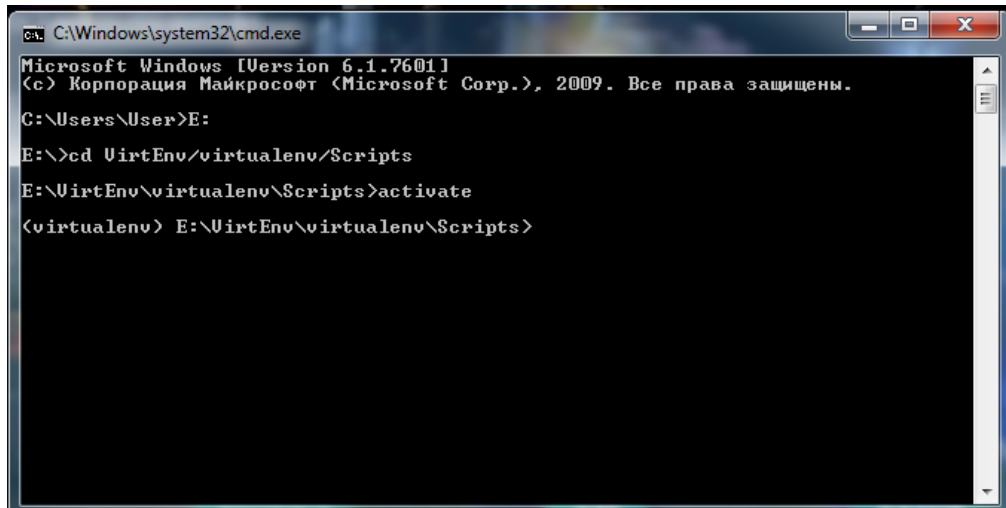
`Include` – C-заголовки, компілюючи пакети Python.

`Lib` – копія версії Python разом з папкою «`site-packages`», в якій встановлена кожна залежність.

`Scripts` – файли, які взаємодіють з віртуальним середовищем.

Установка Django

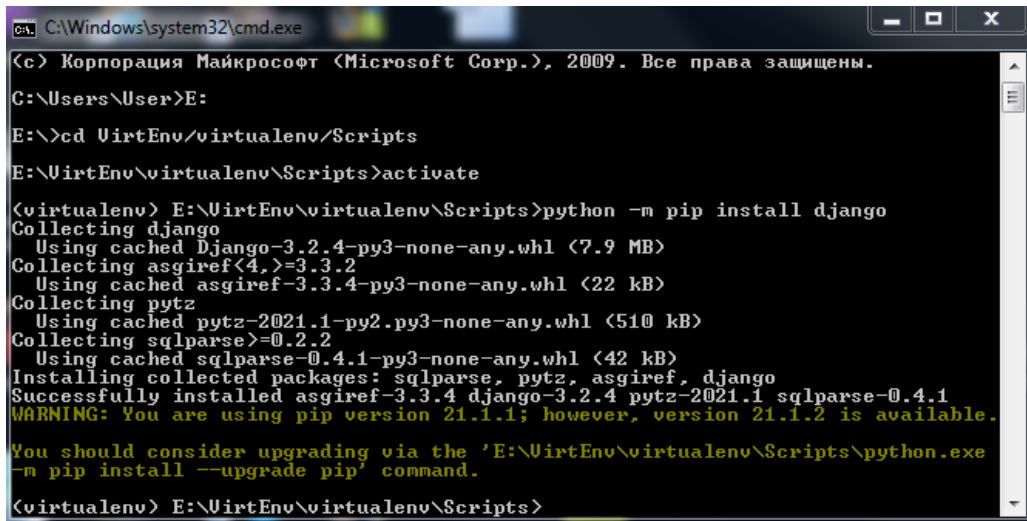
Спочатку активуємо віртуальне середовище за допомогою скрипту activate.bat.



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
(c) Корпорация Майкрософт (Microsoft Corp.), 2009. Все права защищены.
C:\Users\User>E:
E:\>cd UirtEnv\virtualenv\Scripts
E:\UirtEnv\virtualenv\Scripts>activate
(virtualenv) E:\UirtEnv\virtualenv\Scripts>
```

Рис. 2.5. Активация виртуального середовища

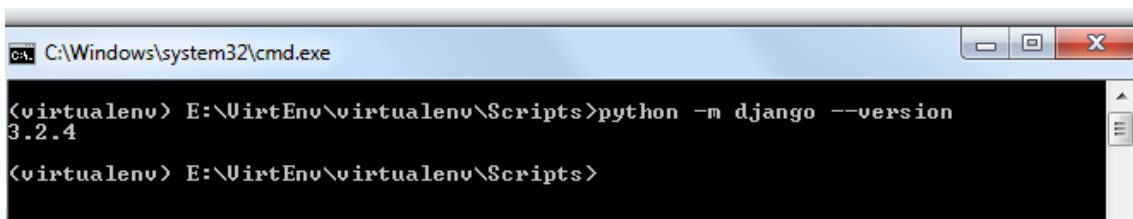
Далі установимо Django за допомогою системи керування пакетами «pip».



```
C:\Windows\system32\cmd.exe
(c) Корпорация Майкрософт (Microsoft Corp.), 2009. Все права защищены.
C:\Users\User>E:
E:\>cd UirtEnv\virtualenv\Scripts
E:\UirtEnv\virtualenv\Scripts>activate
(virtualenv) E:\UirtEnv\virtualenv\Scripts>python -m pip install django
Collecting django
  Using cached Django-3.2.4-py3-none-any.whl (7.9 MB)
Collecting asgiref<4,>=3.3.2
  Using cached asgiref-3.3.4-py3-none-any.whl (22 kB)
Collecting pytz
  Using cached pytz-2021.1-py2.py3-none-any.whl (510 kB)
Collecting sqlparse>=0.2.2
  Using cached sqlparse-0.4.1-py3-none-any.whl (42 kB)
Installing collected packages: sqlparse, pytz, asgiref, django
Successfully installed asgiref-3.3.4 django-3.2.4 pytz-2021.1 sqlparse-0.4.1
WARNING: You are using pip version 21.1.1; however, version 21.1.2 is available.
You should consider upgrading via the 'E:\UirtEnv\virtualenv\Scripts\python.exe
-m pip install --upgrade pip' command.
(virtualenv) E:\UirtEnv\virtualenv\Scripts>
```

Рис. 2.6. Установка Django

Перевіримо успішність установки, виконавши команду для перевірки версії Django.



```

C:\Windows\system32\cmd.exe
(virtualenv) E:\VirtEnv\virtualenv\Scripts>python -m django --version
3.2.4
(virtualenv) E:\VirtEnv\virtualenv\Scripts>

```

Рис. 2.7. Перевірка версії Django

Тестування додатку

Завантажимо сторінку календаря.

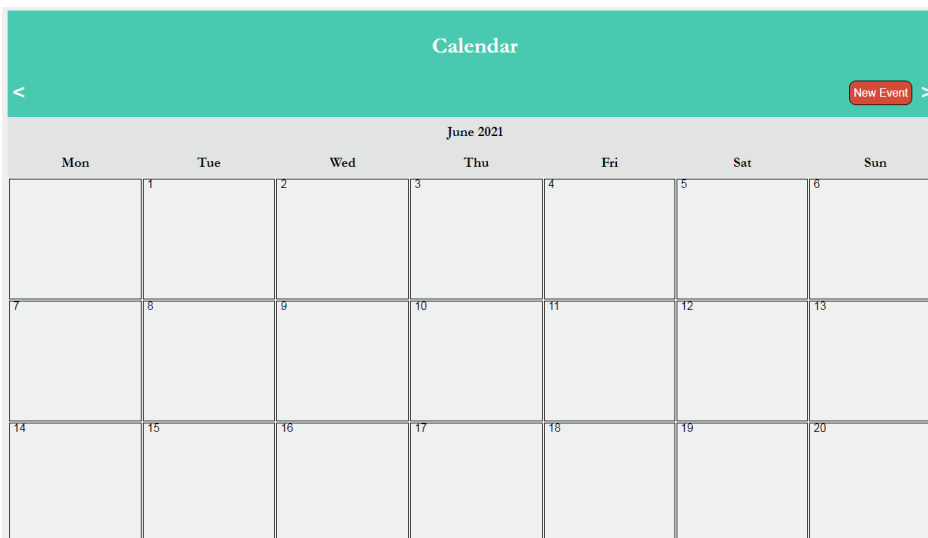


Рис. 2.8. Календарь

На сторінці ми бачимо заголовок календаря, місяць та рік, дні тижня, кнопки «<» та «>» які перемикають місяці, та кнопку «New Event», яка створює нову подію. Створимо за її допомогою нову подію. Натиснувши на кнопку створення події ми переходимо на сторінку створення події.

The image shows a web form titled "Event" with a teal header. In the top left corner, there is a red button labeled "Calendar". The form contains the following fields:

- Title:** A text input field.
- Description:** A large text area.
- Start time:** A date and time picker with a calendar icon on the right.
- End time:** A date and time picker with a calendar icon on the right.

 At the bottom right of the form is a red button labeled "Submit".

Рис. 2.9. Створення події

На сторінці розташована форма для заповнення, кнопка «Calendar» яка повертає нас назад на сторінку за календарем та кнопка «Submit» яка відправляє заповнену форму на сервер, та повертає нас на сторінку з календарем. Всі поля форми повинні бути заповнені, якщо спробувати відправити незаповнену форму, буде показано попередження.

This image shows the same "Event" form as in Figure 2.9, but with a validation error. A yellow warning icon with an exclamation mark is positioned over the "Description" field. A tooltip message in Russian reads "Заполните это поле." (Fill in this field.). The other fields (Title, Start time, End time) and the "Submit" button are visible and appear to be filled or ready for input.

Рис. 2.10 Попередження

Заповнимо та відправимо форму. Поля «Start» та «End time» необхідні для обмеження часових рамок події.

Title: Магазин

Description: Купити молоко, ковбасу та картоплю

Start time: 12.06.2021 13:24

End time: 18.06.2021 13:24

Submit

Рис. 2.11. Заповнена форма

Відправивши форму, нас повернуло на сторінку календаря, на якій з'явилася нова подія.

June 2021						
Mon	Tue	Wed	Thu	Fri	Sat	Sun
	1	2	3	4	5	6
7	8	9	10	11	12 • Магазин	13

Рис. 2.12. Оновлений календар

Подія може мати два кольори в залежності від статусу виконання. Якщо подія не виконана, вона має червоний колір, а якщо виконана – зелений. Якщо натиснути на подію, відкриється сторінка деталей події.

Details	
Calendar	
Name:	Магазин
Description:	Купити молоко, ковбасу та картоплю
Start date:	June 12, 2021, 1:24 p.m.
End before:	June 18, 2021, 1:24 p.m.
Remaining time:	4 days, 2:54:40.987896
Status:	Undone
Edit	Delete Done

Рис. 2.13. Сторінка деталей

На сторінці є два додаткових поля «Remaining time» яке відображає час, який залишився до закінчення події, якщо теперішня дата буде пізніше ніж дата закінчення події, то поле буде мати значення «Expired!». Поле «Status» відображає статус виконання події, може знати два значення «Done» та «Undone». Кнопка «Edit» дозволяє редагувати подію, кнопка «Delete» видаляє подію, а кнопка «Done» змінює статус події. Далі змінимо статус події натиснувши кнопку «Done».

Calendar

Name: Магазин

Description: Купити молоко, ковбасу та картоплю

Start date: June 12, 2021, 1:24 p.m.

End before: June 18, 2021, 1:24 p.m.

Remaining time: 4 days, 2:47:15.330405

Status: Done

[Edit](#) [Delete](#) [Done](#)

Рис. 2.14. Зміна статусу

Як бачимо, поле «Status» змінило своє значення. Тепер відредагуємо подію натиснувши кнопку «Edit».

Calendar

Title:

Description:

Start time:

End time:

[Submit](#)

Рис. 2.15. Змінимо назву події

У результаті змінилася назва події, та колір – через зміну статусу.

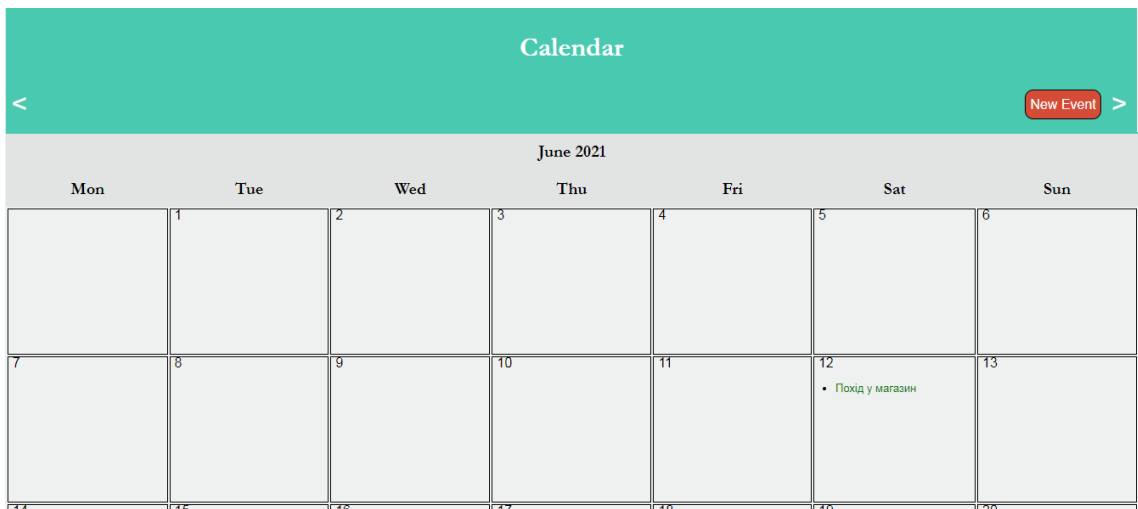


Рис. 2.16. Змінена подія

Висновки

У процесі виконання дипломної роботи були отримані наступні результати:

- Було проаналізовано значення календаря для людства та шлях розвитку календарю.
- Розглянуті особливості МП Python та Django.
- Розроблений веб-додаток календар.

Використання фреймворку Django суттєво економить час та дозволяє сконцентрувати інтелектуальні ресурси для інших задач.

Розробка додатку економічно вигідна, так як використовується безкоштовні засоби розробки. Для використання продукту немає необхідності у спеціальних знаннях або навичках, так як додаток має інтуїтивно зрозумілий інтерфейс.

Список джерел

1. Место и роль календаря в жизни людей. (Электрон. ресурс) Способ доступа: URL: <https://taen-1.livejournal.com/470533.html>
2. Календари в первобытном обществе. (Электрон. ресурс) Способ доступа: URL: <https://w.histrf.ru/articles/article/show/kaliendar>
3. Календарь. История. (Электрон. ресурс) Способ доступа: URL: <https://ru.wikipedia.org/wiki/%D0%9A%D0%B0%D0%BB%D0%B5%D0%BD%D0%B4%D0%B0%D1%80%D1%8C#%D0%98%D1%81%D1%82%D0%BE%D1%80%D0%B8%D1%8F>
4. Черепнин Л.В. Русская хронология. Москва. 1944 год. - 94 с.
5. Емельянов В.В. Ниппурский календарь и ранняя история Зодиака. СПб.: Петербургское востоковедение, 1999 год. - 272 с.
6. Бикерман Э. Хронология древнего мира: Ближний Восток и античность. Москва: Наука, 1975 год. - 336 с.
7. Идельсон Н.И. История календаря. Ленинград: Научное книгоиздательство, 1925 год. - 176 с.
8. Hassig Ross. Time, History, and Belief in Aztec and Colonial Mexico. Austin: University of Texas Press. pp. xv, 220 pp.
9. Селешников С.И. История календаря и хронология. Москва: Наука, 1970 год. - 224 с.
10. История создания языка программирования Python. (Электрон. ресурс) Способ доступа: URL: https://web.informatics.ru/works/17-18/web_online/barabanov_n_v/language_python.html
11. Основы Python. (Электрон. ресурс) Способ доступа: URL: <https://habr.com/ru/post/31180/>
12. История Django. (Электрон. ресурс) Способ доступа: URL: <https://djbook.ru/ch01s03.html>

13. Почему Django – лучший фреймворк для разработки сайта. (Электрон. ресурс) Способ доступа: URL: <https://ru.hexlet.io/blog/posts/pochemu-django-luchshiy-freymvork-dlya-razrabotki-saytov>

ВІДОМОСТІ МАТЕРІАЛІВ ДИПЛОМНОГО ПРОЕКТУ

		Позначення			Найменування	Кільк. аркушів	Примітка		
1									
2					Документація				
3									
4		ІТКІ.КР 19.03.ДА.ПЗ			Пояснювальна записка	92			
5									
6					Презентація	20			
7									
8					Диск CD з презентацією	1			
					ІТКІ.КР 19.03.ДА.ПЗ				
Зм.	Ар- куш	№ докум	Підпис	Дата					
Розроб.		В.С. Крутько			Матеріали кваліфікаційної роботи	Літ.	Аркуш	Арк.	
						Н	1	1	
Керівник		Г.М. Коротенко				НТУ «ДП», 126-17-1			
Рецензент		Л.М. Коротенко							
Н.контр.		Г.М. Коротенко							
Зав.каф.		В.В.Гнатушенко							

Текст програми

Файли додатку Calendar

Файл utils.py

```

from datetime import datetime, timedelta
from calendar import HTMLCalendar
from .models import Event

class Calendar(HTMLCalendar):
    def __init__(self, year = None, month = None):
        self.year = year
        self.month = month
        super(Calendar, self).__init__()

    def formatday(self, day, events):
        events_per_day = events.filter(start_time__day = day)
        ev = ''
        for event in events_per_day:
            ev += f'<li> {event.get_html_url} </li>'
        if day != 0:
            return f"<td><span class = 'date'>{day}</span><ul> {ev} </ul></td>"
        return '<td></td>'

    def formatweek(self, theweek, events):
        week = ""
        for day, weekday in theweek:
            week += self.formatday(day, events)
        return f'<tr> {week} </tr>'

    def formatmonth(self, withyear = True):
        events = Event.objects.filter(start_time__year=self.year, start_time__month=self.month)

        cal = f'<table border = "0" cellpadding = "0" cellspacing = "0" class = "calendar">\n'
        cal += f'{self.formatmonthname(self.year, self.month, withyear=withyear)}\n'
        cal += f'{self.formatweekheader()}\n'
        for week in self.monthdays2calendar(self.year, self.month):
            cal += f'{self.formatweek(week, events)}\n'
        return cal

```

Файл models.py

```

from django.db import models
from django.urls import reverse
from datetime import *

class Event(models.Model):
    title = models.CharField(max_length = 200)
    description = models.TextField()
    start_time = models.DateTimeField()
    end_time = models.DateTimeField()
    status = models.BooleanField(default=False)

    def TimeRemains(self):
        remaining_time = self.end_time - datetime.now(timezone.utc)
        expire_check = self.end_time
        if remaining_time.total_seconds() < 0:
            return 'Event expired!'
        return remaining_time

    @property
    def get_html_url(self):
        url = reverse('Calendar:details', args=(self.id,))

```

```

if self.status == False:
    return f'<a href="{url}" style="color:red;"> {self.title} </a>'
else:
    return f'<a href="{url}" style="color:green;"> {self.title} </a>'

```

Файл views.py

```

from datetime import datetime, timedelta, date
from django.shortcuts import render, get_object_or_404
from django.http import HttpResponse, HttpResponseRedirect
from django.views import generic
from django.urls import reverse
from django.utils.safestring import mark_safe
import calendar

from .models import *
from .utils import Calendar
from .forms import EventForm

class CalendarView(generic.ListView):
    model = Event
    template_name = 'Calendar/calendar.html'

    def get_context_data(self, **kwargs):
        context = super().get_context_data(**kwargs)

        date = get_date(self.request.GET.get('month', None))

        cal = Calendar(date.year, date.month)

        html_cal = cal.formatmonth(withyear=True)
        context['calendar'] = mark_safe(html_cal)

        date = get_date(self.request.GET.get('month', None))
        context['prev_month'] = prev_month(date)
        context['next_month'] = next_month(date)
        return context

    def get_date(required_month):
        if required_month:
            year, month = (int(x) for x in required_month.split('-'))
            return date(year, month, day = 1)
        return datetime.today()

    def previous_month(current_month):
        first = current_month.replace(day=1)
        prev_month = cur_month - timedelta(days=1)
        month = 'month=' + str(prev_month.year) + '-' + str(prev_month.month)
        return month

    def next_month(current_month):
        days_in_month = calendar.monthrange(current_month.year, current_month.month)[1]
        last = current_month.replace(day=days_in_month)
        next_month = last + timedelta(days=1)
        month = 'month=' + str(next_month.year) + '-' + str(next_month.month)
        return month

    def details(request, event_id):
        choosenEvent = Event.objects.get(id = event_id)
        return render(request, 'Calendar/details.html', {'choosenEvent':choosenEvent})

    def new_event(request):
        instance = Event()

```

```

form = EventForm(request.POST or None, instance=instance)
if request.POST and form.is_valid():
    form.save()
    return HttpResponseRedirect(reverse('Calendar:calendar'))
return render(request, 'Calendar/event.html', {'form': form})

def edit_event(request, event_id=None):
    instance = Event()
    if event_id:
        instance = get_object_or_404(Event, pk=event_id)
    form = EventForm(request.POST or None, instance=instance)
    if request.POST and form.is_valid():
        form.save()
        return HttpResponseRedirect(reverse('Calendar:details', args=(event_id,)))
    return render(request, 'Calendar/event.html', {'form': form})

def delete_event(request, event_id):
    event_to_delete = Event.objects.get(id = event_id)
    event_to_delete.delete()
    return HttpResponseRedirect(reverse('Calendar:calendar'))

def status_change(request, event_id):
    event = Event.objects.get(id = event_id)
    event.status = True;
    event.save()
    return HttpResponseRedirect(reverse('Calendar:calendar'))

```

Файл urls.py

```

from django.conf.urls import url
from . import views

app_name = "Calendar"
urlpatterns = [
    url(r'^calendar/$', views.CalendarView.as_view(), name = 'calendar'),
    url(r'^event/new/$', views.new_event, name='event_new'),
    url(r'^event/details/(?P<event_id>(d+)/$', views.details, name='details'),
    url(r'^event/edit/(?P<event_id>(d+)/$', views.edit_event, name='edit_event'),
    url(r'^event/delete/(?P<event_id>(d+)/$', views.delete_event, name='delete_event'),
    url(r'^event/status/(?P<event_id>(d+)/$', views.status_change, name='change_status'),
]

```

Файл forms.py

```

from django.forms import ModelForm, DateInput

from .models import Event

class EventForm(ModelForm):
    class Meta:
        model = Event
        widgets = {
            'start_time': DateInput(attrs={'type': 'datetime-local'}, format='%Y-%m-%dT%H:%M'),
            'end_time': DateInput(attrs={'type': 'datetime-local'}, format='%Y-%m-%dT%H:%M'),
        }
        fields = ['title','description','start_time','end_time']

    def __init__(self, *args, **kwargs):
        super(EventForm, self).__init__(*args, **kwargs)
        self.fields['start_time'].input_formats = ('%Y-%m-%dT%H:%M',)
        self.fields['end_time'].input_formats = ('%Y-%m-%dT%H:%M',)

```

Файли стилів

Файл style.css

```
body {
  font-family: "ptsans", "Lato", "Helvetica Neue", Helvetica, Arial, sans-serif;
  background: #EFF0F0;
  z-index: -2;
}

.title {
  text-align: center;
  margin: 0px;
  background: #1abc9c;
  height: 100px;
  position: relative;
  width: 100%;
}

.coverColor {
  position: absolute;
  background: #48C9B0;
  height: 2px;
  width: 99.15%;
}

.coverColor2 {
  position: absolute;
  background: #E2E3E3;
  height: 85px;
  top: 160px;
  width: 99.15%;
  z-index: -1;
}

.PageTitle {
  display: flex;
  justify-content: center;
  align-items: center;
  background: #48C9B0;
  height: 100px;
  color: white;
  font-family: Garamond;
}

/* buttons styles */

.left {
  float: left;
}

.right {
  float: right;
}

.btn {
  padding: 7px 5px;
  color: #fff;
}
```

```
background: rgb(212,75,56);
border: 1.5px solid black;
border-radius: 10px;
box-shadow: 0 0 0 0;
margin-right: 5px;
text-decoration: none;
outline: none;
}

.btn:hover {
background: rgb(232,95,76);
}

.btn:active {
background: rgb(152,15,0);
}

.month-button {
font-size: 200%;
position: relative;
bottom: 3px;
background: none;
border-color: #48C9B0;
padding-top: 0px;
padding-bottom: 0px;
}

.clearfix {
margin: 0px;
background: #48C9B0;
position: relative;
height: 50px;
}

/* Form styles */
.form {
margin: auto;
}

.form input, .form select, .form textarea {
border-radius: 5px;
border: 1px solid #17a2b8;
outline: none;
background: none;
padding: 5px;
width: 100%;
}

/* App styles */

.calendar {
width: 100.12%;
```

```

font-size: 13px;
}

.calendar tr, .calendar td {
border: 1px solid black;
}

.calendar th {
padding: 10px;
text-align: center;
font-family: Garamond;
font-size: 20px;
background: #E2E3E3;
}

.calendar td {
width: 200px;
height: 150px;
padding: 20px 0px 0px 5px;
}

.month {
font-size: 25px;
background: #E2E3E3;
}

.date {
font-size: 16px;
}

ul {
height: 100%;
padding: 0px 5px 0px 20px;
}

a {
text-decoration: none;
}

```

Файл details.css

```

section {
display: inline-block;
margin: 20px;
}

.display {
display: inline;
}

.detailsMenu {
width: 100%;
position: relative;
height: 50px;
margin: 0px;
}

```

```
.detailButton {
    height: 15px;
    width: 50px;
    text-align: center;
    margin-left: 10px;
}

.DetailTitle {
    font-family: Arial;
    font-weight: bold;
}

.DetailContent {
    font-family: Garamond sans-serif;
    margin-left: 10px;
}
```

Файли шаблонів

Calendar.html

```
{% extends 'base.html' %}

{% block styles %}{% endblock %}

{% block title %}
<div class="PageTitle">Calendar</div>
{% endblock %}

{% block content %}

<div class="clearfix">
    <a class="btn left month-button" href="{% url 'Calendar:calendar' %}"?{{ prev_month }}"></a>
    <a class="btn right month-button" href="{% url 'Calendar:calendar' %}"?{{ next_month }}"></a>
    <a class="btn right" href="{% url 'Calendar:event_new' %}"> New Event </a>
</div>
<div class="coverColor"></div>
<div class="coverColor2"></div>

{{ calendar }}

{% endblock %}
```

Details.html

```
{% extends 'base.html' %}

{% load static %}

{% block styles %}
<link rel="stylesheet" href="{% static 'assets/css/details.css' %}">
{% endblock %}

{% block title %}
<div class="PageTitle">Details</div>
{% endblock %}

{% block content %}

<div class="clearfix">
    <a class="btn left" href="{% url 'Calendar:calendar' %}"> Calendar </a>
</div>

<section>
    <p class="display DetailTitle">Name:</p><p class="display DetailContent">{{ chosenEvent.title }}</p>
</section>
<br><hr>
```

```

<section>
  <p class="display DetailTitle">Description:</p><p class="display DetailContent">{{ chosenEvent.description }}</p>
</section>
<br><hr>
<section>
  <p class="display DetailTitle">Start date:</p><p class="display DetailContent">{{ chosenEvent.start_time }}</p>
</section>
<br><hr>
<section>
  <p class="display DetailTitle">End before:</p><p class="display DetailContent">{{ chosenEvent.end_time }}</p>
</section>
<br><hr>
<section>
  <p class="display DetailTitle">Remaining time:</p><p class="display DetailContent">{{ chosenEvent.TimeRemains }}</p>
</section>
<br><hr>
<section>
  {% if chosenEvent.status == False %}
    <p class="display DetailTitle">Status:</p>
    <p class="display DetailContent">Undone</p>
  {% elif chosenEvent.status == True %}
    <p class="display DetailTitle">Status:</p>
    <p class="display DetailContent">Done</p>
  {% endif %}
</section>
<hr>
<div class="detailsMenu">
  <a class="btn detailButton left" href="{% url 'Calendar:edit_event' event_id=chosenEvent.id %}"> Edit </a>
  <a class="btn detailButton left" href="{% url 'Calendar:delete_event' event_id=chosenEvent.id %}"> Delete </a>
  <a class="btn detailButton left" href="{% url 'Calendar:change_status' event_id=chosenEvent.id %}"> Done </a>
{% endblock %}

```

Event.html

```

{% extends 'base.html' %}

{% block styles %}{% endblock %}

{% block title %}
<div class="PageTitle">Event</div>
{% endblock %}

{% block content %}
<div class="clearfix">
  <a class="btn left" href="{% url 'Calendar:calendar' %}"> Calendar </a>
</div>

{% if form.errors %}
  {% for field in form %}
    {% for error in field.errors %}
      <div class="alert alert-danger">
        {{ field.label }} <strong>{{ error|escape }}</strong>
      </div>
    {% endfor %}
  {% endfor %}
  {% for error in form.non_field_errors %}
    <div class="alert alert-danger">
      {{ field.label }} <strong>{{ error|escape }}</strong>
    </div>
  {% endfor %}
{% endif %}

<form method="post">
  {% csrf_token %}
  <table class="form form-table">
    {{ form }}

```



```
<tr><td colspan="2"><button type="submit" class="btn btn-info right"> Submit </button></td></tr>
</table>
</form>
{% endblock %}
```

ВІДГУК

на кваліфікаційну роботу бакалавра на тему:
**«Розробка сервісу "Календар" з функціями todo-аркуша
на основі фреймворку Django»**
студента групи 126-17-1 Крутько Владислава Сергійовича.

Метою даної кваліфікаційної роботи є розробка сервісу "Календар" з функціями todo-аркуша на основі фреймворку Django.

Дана тема актуальна тому, що в умовах надшвидкого плину часу кількість одночасно виконуваних та проектуємих людиною справ може сягати декількох десятків і більше. Однією з найновітніших технологій сьогодення є так звані To-do lists (To-do аркуші). To-Do List, це інформаційний простір в якому користувач може створювати нові списки справ (To-Dos), викреслювати зроблені пункти або видаляти їх, зберігати стан списку справ, щоб повернутися до нього пізніше, а також видаляти список справ цілком і багато чого ще.

Така тема кваліфікаційної роботи безпосередньо пов'язана з об'єктом діяльності фахівця галузі знань 12 «Інформаційні технології» спеціальності 126 «Інформаційні системи та технології» – розробка та експлуатація різнопланових інформаційних систем і пов'язаних з ними програмних засобів.

Завдання кваліфікаційної роботи (Розробка сервісу "Календар" з функціями todo-аркуша на основі фреймворку Django) віднесені в освітньо-професійній програмі підготовки випускника відповідної спеціальності до класу евристичних, рішення яких заснована на знаково-розумових умінь фахівця.

Оригінальність технологічних рішень полягає в використанні високорівневого відкритого Python-фреймворка (програмного каркасу) для розробки веб-систем та веб сервісів.

Практичне значення результатів кваліфікаційної роботи полягає в створенні сервісу "Календар" з функціями todo-аркуша на основі фреймворку Django.

Оформлення графічних та текстових матеріалів пояснювальної записки кваліфікаційної роботи виконано на досить високому рівні і без відхилень від стандарту.

Ступінь самостійності виконання кваліфікаційної роботи повною мірою відповідає першому освітньо-кваліфікаційному рівню вищої освіти, тобто ступеню бакалавра.

В роботі досить повно виконано огляд сучасного стану ринку програмних додатків в сфері створення бізнес-рішень відповідного рівня.

Деякі дискусійні положення та несуттєві недоліки пов'язані з не досить чітким викладенням та описом структури роботи та дещо невисокою якістю ілюстративних матеріалів.

Незважаючи на вищевказані зауваження, кваліфікаційна робота в цілому заслуговує оцінки «відмінно», а її виконавець, студент Крутько В.С., присвоєння йому відповідної кваліфікації.

**Керівник кваліфікаційної роботи, професор
кафедри ІТКІ, д-р техн. наук**

Г.М. Коротенко

Рецензія

на кваліфікаційну роботу бакалавра на тему:
**«Розробка сервісу "Календар" з функціями todo-аркуша
на основі фреймворку Django»**
студента групи 126-17-1 Крутько Владислава Сергійовича.

Стрімке зростання обсягу інформаційних потоків, а також пов'язаних з використанням великої кількості інформаційних технологій та підтримуючих їхнє функціонування гаджетів, віджетів, платформ та програмних додатків ставить перед користувачами складні завдання обліку та вирішення відповідної кількості задач.

В рецензованій кваліфікаційній роботі створено користувальницький сервіс "Календар" з функціями todo-аркуша на основі фреймворку Django.

Використовувані технології розробки даних програмних засобів обліку та контролю різноманітних процесів є евристичною компонентою і безпосередньо пов'язане з об'єктом діяльності фахівця спеціальності 126 «Інформаційні системи та технології» галузі знань 12 «Інформаційні технології».

Студент В.С. Крутько досить добре розібралася в специфіці застосування високорівневого Python-орієнтованого веб-фреймворка, який дозволяє швидко створювати безпечні, надійні і легко підтримувані веб-сайти, названого так на честь Джанго Рейнхардта – французького джазового гітариста-віртуоза, одного з засновників стилю «циганський джаз».

До недоліків роботи слід віднести:

- деякі неточності у оформленні роботи;
- дещо спрощено підхід щодо реалізації функцій збору, аналізу та візуалізації результатів формування та обробки даних, необхідних для обробки контенту.

Однак, розроблений сервіс можна вважати закінченим етапом виконаної кваліфікаційної роботи.

На підставі вищевикладеного, можна зробити висновок, що представлені матеріали цілком відповідають вимогам, що пред'являються до кваліфікаційних робіт першого рівня вищої освіти, тобто ступеню бакалавра.

З огляду на весь спектр створених компонентів, що забезпечують формування даної кваліфікаційної роботи, в цілому вона заслуговує на оцінку «відмінно», а її виконавець, студент Крутько В.С., присвоєння йому відповідної кваліфікації.

**Рецензент, доцент кафедри
ПЗКС, канд. техн. наук**

(посада, звання)

Л.М. Коротенко

(підпис)

(прізвище, ініціали)