

УДК 004

РАЗРАБОТКА СИСТЕМЫ ПОДСЧЁТА ДОЗИРОВКИ СЛОЖНЫХ УДОБРЕНИЙ НА ОСНОВЕ ТЕХНОЛОГИЙ ЯЗЫКА PYTHON И БД SQLITE

Карасев В.В, студент гр. ИТб-181, IV курс
Ванеев О.Н, доцент (к.н.) кафедры ИиАПС
Кузбасский государственный технический университет
имени Т. Ф. Горбачева г. Кемерово

В настоящее время актуальной задачей является повышение эффективности функционирования предприятий сельскохозяйственного сектора за счёт привлечения преимуществ современных информационных технологий.

Целью данной разработки является автоматизация и облегчение подсчетов внесения удобрения предприятием для определенного вида зерновой, бобовой и другого вида культуры. С помощью данной программы любое фермерское предприятие сможет избежать излишков затрат на химические и биологические препараты, используемые при удобрении почв или же избавления ее от вредителей.

На данный момент времени большинство КФХ осуществляют подсчеты вручную. Очень часто осуществляются ошибки из-за человеческого фактора при расчёте концентраций действующих веществ, что неминуемо приводит к потерям урожая. В связи с повышением цен на удобрения и химические препараты, актуальность автоматизации расчетов действующих веществ набирает обороты.

Программа разрабатывается на основе актуальных препаратов на сегодняшний день, которые будут храниться в базе данных, на основе этой БД будет реализовано:

- Подсчет общего количества действующего вещества на площадь
- Подсчет необходимых действующих веществ для активной жизнедеятельности посевной культуры для заданной площади.
- Вывод информации о нормах по ГОСТу. (Опять же расчет на выбранную пользователем площадь)

Данная разработка будет осуществляться на базе языка Python и связываться с SQLite Базой данных. Чтобы выводить расчеты из БД, подключим к языку библиотеку интерфейсов Tkinter и осуществим импорт интересующих нас форм. Стоит отметить, что веществ в базу данных можно добавить достаточно много.

Отообразим работу нашей программы на диаграмме деятельности:

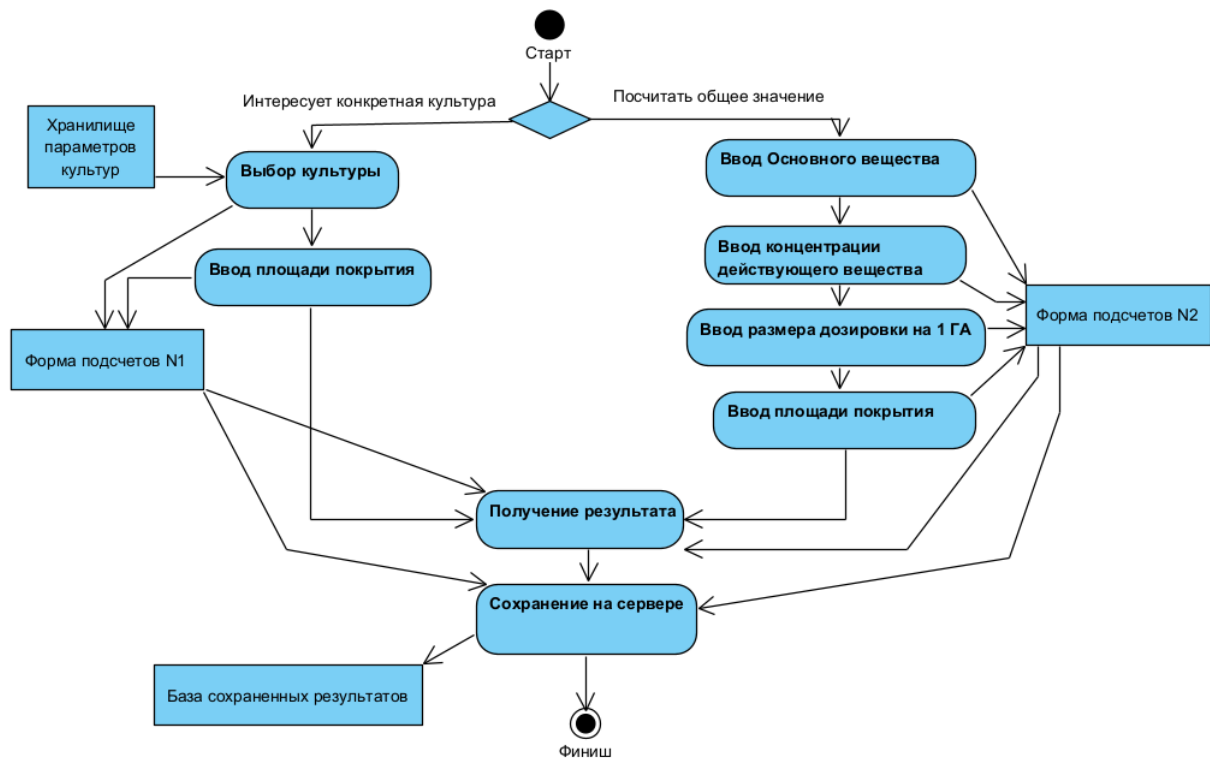


Рисунок 1 – Диаграмма деятельности процесса расчетов

Нашу работу осуществим на платформе PyCharm которая открыта и находится в свободном доступе в интернете.

Ссылка на программу - <https://www.jetbrains.com/pycharm/download/#section=windows>



Version: 2021.3.3
Build: 213.7172.26
17 March 2022

[System requirements](#)
[Installation instructions](#)
[Other versions](#)

Download PyCharm

[Windows](#) [macOS](#) [Linux](#)

Professional

For both Scientific and Web Python development. With HTML, JS, and SQL support.

[Download](#)

Free trial

Community

For pure Python development

[Download](#)

Free, built on open-source

Рисунок 2 – Сайт для скачивания PyCharm

После скачивания и установки открываем и создаем новый проект

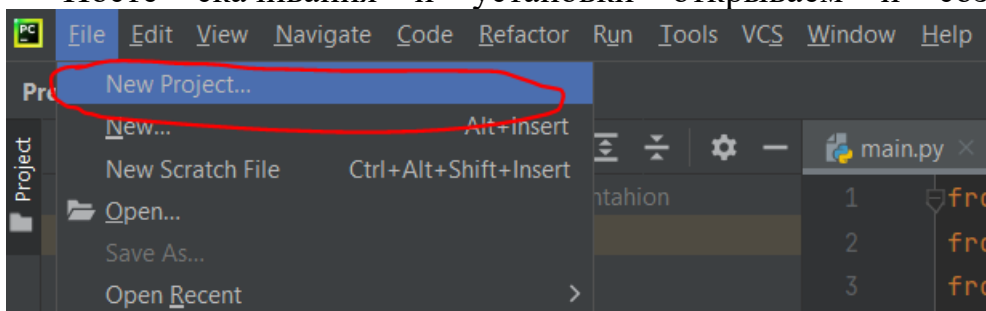


Рисунок 3 – Меню открытия нового проекта

Для создания проекта нужно будет указать несколько параметров,

Во вкладке «Location» будет путь в конце, которого нам надо будет задать имя файла. В данном случае это python Project (Это название папки, в которой будет храниться наш проект).

Далее в след. вкладке «Location» выберем имя нашего файла в котором будем работать(venv).

В основном это все, можно менять типы подключаемых служб во вкладке «New environment using». Но это не обязательно.

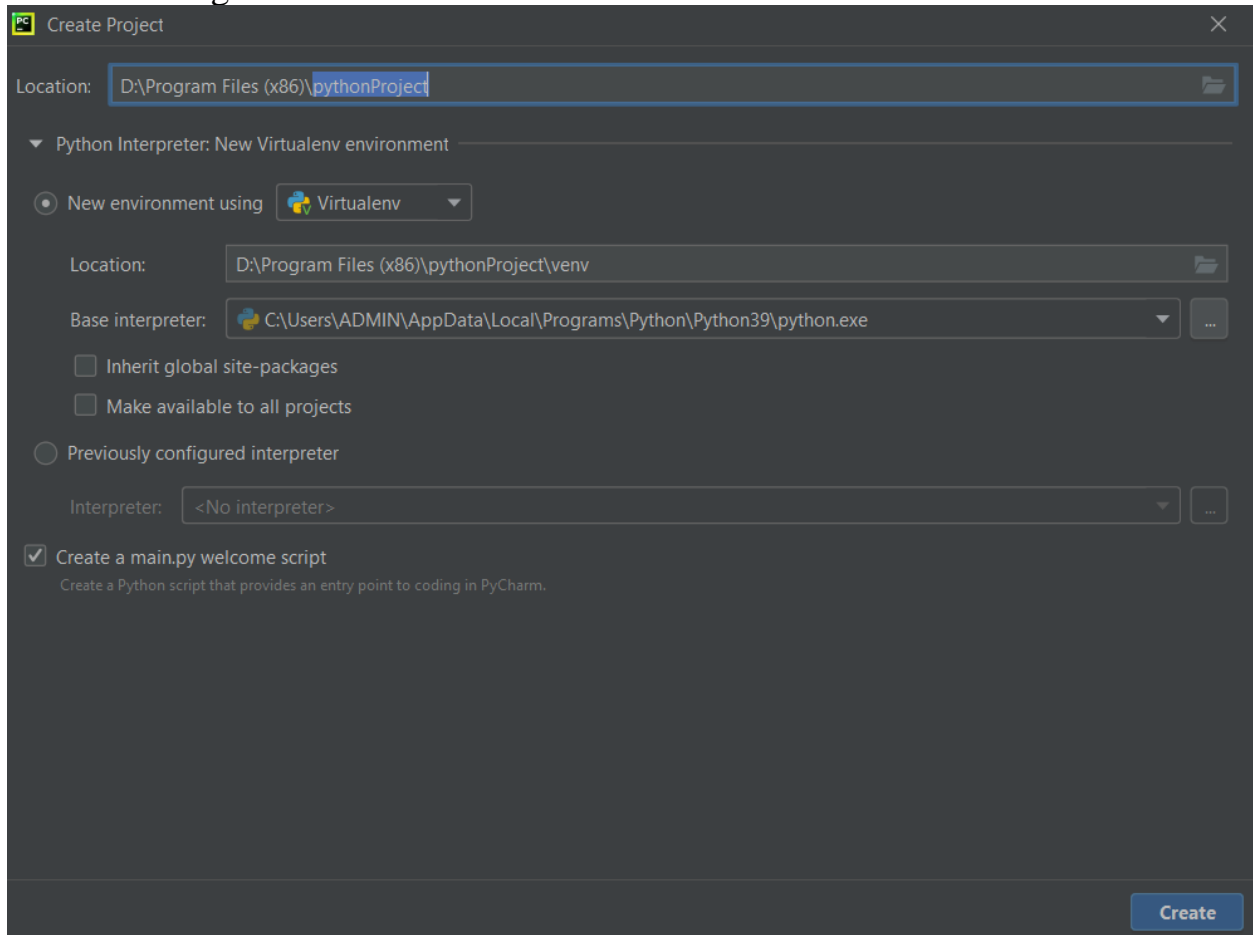


Рисунок 4 – Меню создания проекта

Файл создан, но это еще не все, подключим библиотеки с помощью которых мы будем работать с интерфейсом b с базой данных в SQLite, так же добавим время в систему

```
from tkinter import *  
from tkinter import ttk  
from tkinter.ttk import Combobox  
import sqlite3  
import time
```

Рисунок 5 – Подключение библиотек

Некоторые библиотеки которые нас интересуют приходится дополнительно устанавливать, это делается через нижний терминал с помощью команды:

pip install (Имя_библиотеки)

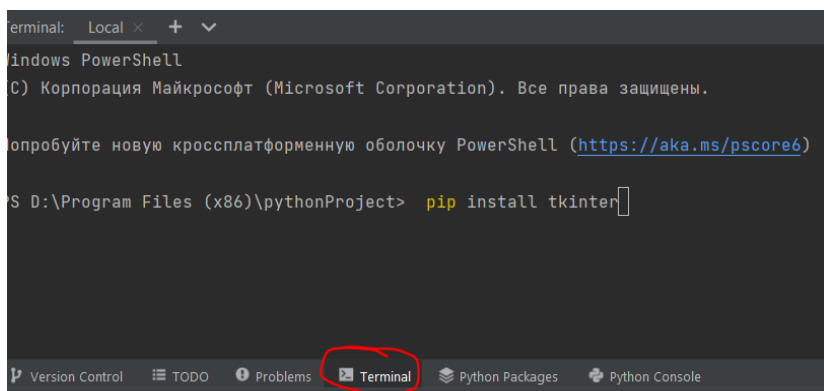


Рисунок 6 – Пример синтаксиса установки дополнительных Библиотек.

Настроили проект, подключили библиотеки. Приступим к выполнению задач, для начала создадим графический интерфейс, который будет выглядеть следующим образом:

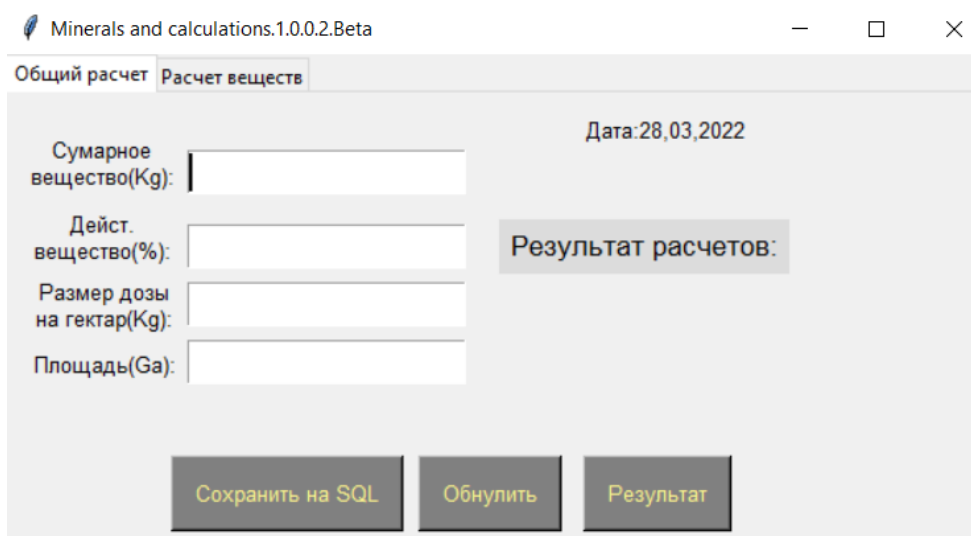


Рисунок 7 – Вид первой вкладки графического интерфейса.

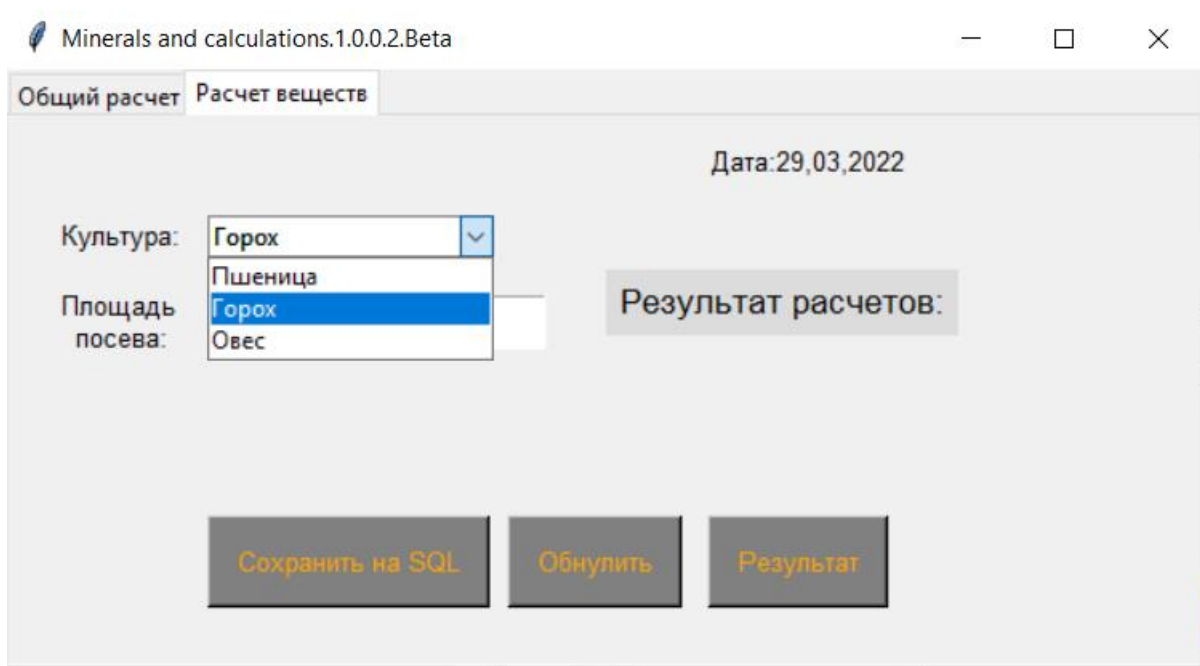


Рисунок 8 – Вид второй вкладки графического интерфейса.

На основе таких элементов как :

Entry – поле ввода

```
Doza = Entry(tab1, width=15, font=10)
Doza.place(x=110, y=115)
```

Рисунок 9.1 – Код для создания нужного поля ввода

Label – Текстовая подпись

```
lbl7 = Label(tab2, text = "Площадь \nпосева:", font=("Arial Bold", 10),
             padx="5", pady="5")
lbl7.place(x=20, y=80)
```

Рисунок 9.2 – Код для создания нужного текста для обозначения

Button - кнопка

```
Chet1 = Button(tab1, text="Результат", padx="10", pady="10",
               font=("Arial Bold", 10), bg="gray", fg="khaki1", command=chet)
Chet1.place(x=350, y=220)
```

Рисунок 9.3 – Код для создания Кнопки

Мы создали нужную нам форму. Далее создадим процедуры, которые осуществят нужные нам подсчеты.

```
def chet():
    SummW = float(Summ_wesh.get())
    Deist = float(Deistv_vesh.get())
    c = (SummW / 100) * Deist
    d = str(round(c, 2))
    f = str(SummW)

    Doza_na_ga = float(Doza.get())
    Doza_na_S = float(Sploshad.get())
    Proporh = Doza_na_ga * Doza_na_S
    F = (Proporh * SummW) / c

    Gaa = str(Doza_na_S)
    FF = str(round(F, 2))
    lbl10.configure(text="Результат расчетов:" + "\nВ " + f + " Kg Д/в (Kg):" + d + "\nНа " + Gaa + "Га понадобится " + FF + "Kg")
```

Рисунок 10.1 – Код работы процедуры chet()

```
def Func_Shet():  
    if Plant.get() == "Пшеница":  
        S = float(Sploshad2.get())  
        Azot = float(47.8)  
        Fosfor = float(18.8)  
        Kalii = float(31.5)  
        summA = str(Azot * S)  
        summB = str(Fosfor * S)  
        summC = str(Kalii * S)  
  
        lbl10.configure(  
            text="Результат расчетов(На" + str(S) + ")Га:" +  
            "\n Азота необходимо " + summA +  
            "Kg" + "\n Фосфора необходимо " + summB + "Kg" + "\n Калия необходимо " + summC + "Kg")
```

Рисунок 10.2 – Код работы процедуры Funs_Shet()

Перейдем к результату работы наших процедур.

Minerals and calculations.1.0.0.2.Beta

Общий расчет | Расчет веществ

Дата: 28,03,2022

Суммарное вещество(Kg): 50

Действ. вещество(%): 10

Размер дозы на гектар(Kg): 0.4

Площадь(Ga): 30

Результат расчетов:
В 50.0 Kg Д/в (Kg):5.0
На 30.0Ga понадобится 120.0Kg

Сохранить на SQL | Обнулить | Результат

Рисунок 11.1 – Осуществление и связь функции chet() с интерфейсом.

Рисунок 11.2 – Осуществление и связь функции Funs_Shet() с интерфейсом.

Так же у нас существует выгрузка результата и вводимых параметров в SQLite. Так выглядит процедура выгрузки в которой мы добавим значения в нашу таблицу на SQL.

```

8
9  def Save_sql():
10
11      SummW = float(txt.get())
12      Deist = float(txt1.get())
13      c = (SummW / 100) * Deist
14      Doza_na_ga = float(txt2.get())
15      Doza_na_S = float(txt3.get())
16      Proporh = Doza_na_ga * Doza_na_S
17      F = (Proporh * SummW) / c
18      ID = 1
19
20      with sqlite3.connect('Ydobrenie.db') as db:
21          WOrk_body = db.cursor()
22          query = """INSERT INTO Dannie_ferms (ID, Osn_Vehestvo, Procent_aktiv_Vesh, Doza_Na_Ga,
23          S_ploshad, Soderhanie_aktiv_Vesh, Neobhod-veshestvo, Data_time) VALUES(?,?,?,?,?,?,?); """
24          result = WOrk_body.fetchall()
25          for x in result:
26              ID = ID + 1
27              print(ID)
28          Peremennaya = (ID, SummW, Deist, Doza_na_ga, Doza_na_S, Proporh, F, time_str)
29          WOrk_body.execute(query, Peremennaya)
30          db.commit()
31          WOrk_body.close()

```

Результат нажатия на кнопку «Сохранить в SQL»
Данные сохранились на сервере в базе данных «Ydobrenie.db»

DB Browser for SQLite - D:\Program Files (x86)\pythonProject1\Ydobrenie.db

Файл Редактирование Вид Инструменты Справка

Новая база данных Открыть базу данных Записать изменения Отменить изменения Открыть проект Сохранить проект Прикрепить БД Закр

Структура БД Данные Прагмы SQL

Таблица: Dannie_ferms Filter in any column

ID	Osn_Veshstvo	Procent_aktiv_Vesh	Doza_Na_Ga	S_ploshad	Soderhanie_aktiv_Vesh	Neobxod_veshstvo	Data_time
Фи...	Фильтр	Фильтр	Фильтр	Фильтр	Фильтр	Фильтр	Фильтр
1	1	50	10	0.4	30	5	120 27,03,2022

Список литературы

1. **Лутц М. Изучаем Python**, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 1280 с.
2. **Златопольский Д.М. Основы программирования на языке Python**. – М.: ДМК Пресс, 2017. – 284 с.
3. **Лутц М. Программирование на Python**, том I, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 992 с.
4. **Лутц М. Программирование на Python**, том II, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 992 с.
5. **Гэддис Т. Начинаем программировать на Python**. – 4-е изд.: Пер. с англ. – СПб.: БХВ-Петербург, 2019. – 768 с.