**Этап 1. Подготовка ПО к работе:**

Перед началом работы необходимо установить интерпретатор языка Python 3.7.1 версии: <https://www.python.org/downloads/windows/>

Затем среду программирования JetBrains PyCharm: (Community): <https://www.jetbrains.com/pycharm/download>

Далее необходимо загрузить ODBC драйвер: <https://www.microsoft.com/en-us/download/details.aspx?id=56567>

Устанавливаем коннектор MySQL с Python по ссылке: <https://dev.mysql.com/downloads/connector/python/>

Затем открываем командную строку и прописываем:

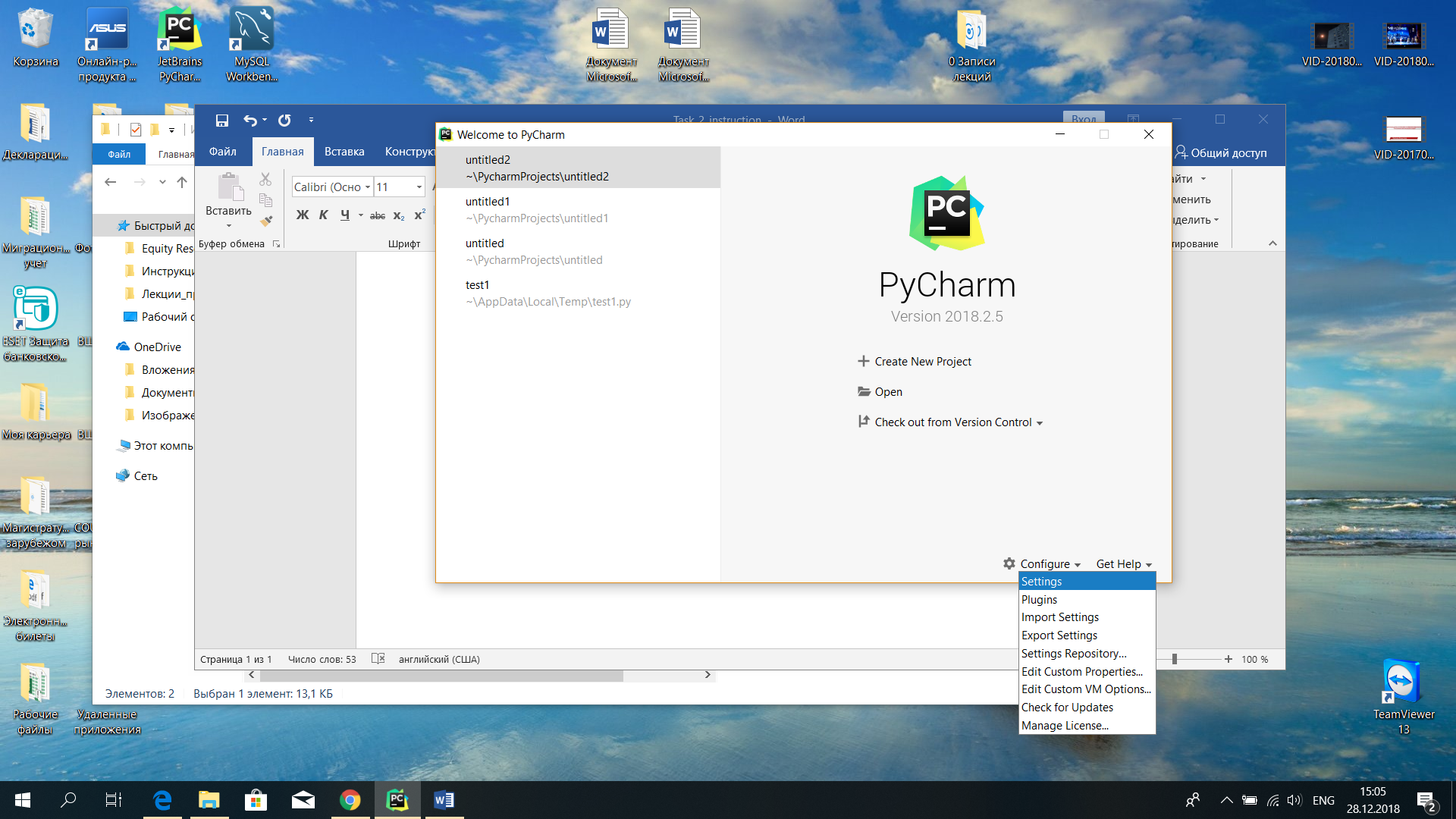
>pip install pyodbc

>pip install mysql-connector-python

Открываем PyCharm идем в PyCharm

Через Configure –> Settings -> Project Interpreter настраиваем необходимые пакеты:

pip, matplotlib, numpy, mysql-connector, prettytable



**Этап 2. Подключение Python к MySQL:**

import mysql.connector  
import numpy as np  
  
cnx = mysql.connector.connect(user='root', password='psw1612', database='dbaseorderlog')  
cursor = cnx.cursor()

**Этап 3. Работа с Python:**

***Предложение пользователю ввести тикер и время:***

print('Введите Тикер:')  
SEC\_CODE = input()  
  
print('Введите время (9 знаков):')  
timestamp = input()

***Формирование запроса по введенным данным о потоке заявок по указанному тикеру до указанного момента времени:***

query1 = f'select PRICE, VOLUME from Orderlog where SECCODE="{SEC\_CODE}" and BUYSELL="S" and time <={timestamp} ' \  
 f'and PRICE > 0 group by PRICE order by PRICE desc'  
cursor.execute(query1)  
result1 = cursor.fetchmany(15000)  
print('Поток заявок на продажу по тикеру',result1)  
  
query2 = f'select PRICE, VOLUME from Orderlog where SECCODE="{SEC\_CODE}" and BUYSELL="B" and time <={timestamp} ' \  
 f'and PRICE > 0 group by PRICE order by PRICE ASC'  
cursor.execute(query2)  
result2 = cursor.fetchmany(15000)  
print('Поток заявок на покупку по тикеру',result2)

***Выделение 20 лучших заявок на покупку и продажу для формирования стакана и отображения диаграммы:***

best\_asks = sorted(result1, key=lambda n: n[0])[:20]  
best\_bids = sorted(result2, key=lambda n: -n[0])[:20]  
print('best\_bids:', best\_bids)  
print('best\_asks:', best\_asks)

***Отображение стакана:***

from prettytable import PrettyTable  
Order\_book = PrettyTable()  
Bid\_price, Bid\_volume = zip(\*best\_bids)  
Ask\_price, Ask\_volume = zip(\*best\_asks)  
Order\_book.add\_column("Bid volume", Bid\_volume)  
Order\_book.add\_column("Bid\_price", Bid\_price)  
Order\_book.add\_column("Ask\_price", Ask\_price)  
Order\_book.add\_column("Ask\_volume", Ask\_volume)  
print(Order\_book)

***Отображение диаграммы:***

import matplotlib.pyplot as plt  
plt.bar(\*zip(\*best\_asks), label='Ask', color='red')  
plt.bar(\*zip(\*best\_bids), label='Bid', color='green')  
plt.ylabel('Volume')  
plt.xlabel('Price')  
plt.title('Depth of Market')  
plt.legend(loc='upper right')  
plt.show()

**Этап 4. Ожидаемые результаты:**

