

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені Тараса Шевченка
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра програмних систем і технологій
Дисципліна
«Якість програмного забезпечення та тестування»
Лабораторна робота №11
на тему
«Покрокове висхідне тестування»

Виконав:	Мішак Максим		Перевірила:	Карнаух Тетяна Олександрівна
Група	ІПЗ-33		Дата перевірки	
Форма навчання	денна		Оцінка	
Спеціальність	121			
2023				

Завдання :

Створити файл, що містить відомості про наявність квитків на рейсах автовокзалу. Структура даних: номер рейсу, пункт призначення, час вирушення, ціна квитка, кількість вільних місць.

Написати програму, що видає наступну інформацію

- час відправлення автобусів до міста X;
- наявність вільних місць та номер рейсу до міста X; Кожну з функцій розмістити в окремому заголовочному файлі.

Виконання :

Для виконання лабораторної роботи було створено датасет , в якому зібрано інформацію про ринкові перевезення з Києва до відповідних населених пунктів . Таблиця має наступний вигляд :

flight_table

flight_number	destination	departure_time	available_places
A32	Prague	25.04.2023	10
X91	Paris	04.05.2023	1
B01	Amsterdam	23.04.2023	14
A70	Vienna	24.04.2023	2
C50	Berlin	23.04.2023	5
X13	London	01.05.2023	20
D77	Grindelwald	27.04.2023	3
C22	Strasbourg	29.04.2023	9
A95	Florence	12.05.2023	15
B16	Salzburg	25.04.2023	1
X43	Rome	20.05.2023	12
C12	Dublin	08.05.2023	6
D01	Copenhagen	04.05.2023	10
A88	Budapest	05.05.2023	5
B29	Oslo	26.04.2023	10
X11	Barcelona	29.05.2023	21
C01	Venice	03.05.2023	13
D99	Cappadocia	08.05.2023	9
A10	Stockholm	30.04.2023	8
X99	Munich	29.04.2023	11

Для її обробки відповідним чином було написано застосунок на мові програмування Python .

Код застосунку :

main.py

```
main.py x
1  from AvailablePlaces import AvailablePlaces
2  from DepartureTime import DepartureTime
3
4  print('Choose the action please . Available actions : ')
5  print('1)Departure time(you should choose the city)')
6  print('2)Available places(you should choose the city)')
7  chosenAction = input('Action : ')
8  departmentTime = DepartureTime()
9  availablePlaces = AvailablePlaces()
10 if chosenAction == '1':
11     departmentTime.getByTheCity()
12 elif chosenAction == '2':
13     availablePlaces.getByTheCity()
14 else:
15     while chosenAction != '1' or chosenAction != '2':
16         print('Wrong action')
17         chosenAction = input('Action : ')
18
```

DepartureTime.py

```
DepartureTime.py x
1  import pandas as pd
2
3
4  class DepartureTime:
5      @staticmethod
6      def getByTheCity():
7          df = pd.read_csv('flight_table.csv')
8          print('Choose the city : ')
9          destination = df['destination'].values
10         print(destination)
11         chosenCity = input()
12         while chosenCity not in destination:
13             chosenCity = input('Undefined city !')
14         else:
15             departureTime = df[df['destination'] == chosenCity]['departure_time'].values
16             print(f'Departure time to {chosenCity} : {departureTime}')
```

AvailablePlaces.py

```
1 import pandas as pd
2
3
4 class AvailablePlaces:
5     @staticmethod
6     def getByTheCity():
7         df = pd.read_csv('flight_table.csv')
8         print('Choose the city : ')
9         destination = df['destination'].values
10        print(destination)
11        chosenCity = input()
12        while chosenCity not in destination:
13            chosenCity = input('Undefined city !')
14        else:
15            departureTime = df[df['destination'] == chosenCity]['available_places'].values
16            flightNumber = df[df['destination'] == chosenCity]['flight_number'].values
17            print(f'Available places for flight number {flightNumber} : {departureTime}')
```

Тестові варіанти для сторонніх модулів :

DepartureTime.py

- TB1 (Тест на правильність роботи функції для випадку, коли вхідні дані відсутні.)
 - Очікуваний результат - вивід повідомлення “Undefined city” та активне поле для вводу інформації
 - Результат роботи програми :

```
Choose the city :
['Prague' 'Paris' 'Amsterdam' 'Vienna' 'Berlin' 'London' 'Grindelwald'
 'Strasbourg' 'Florence' 'Salzburg' 'Rome' 'Dublin' 'Copenhagen'
 'Budapest' 'Oslo' 'Barcelona' 'Venice' 'Cappadocia' 'Stockholm' 'Munich']
Undefined city !
```

- TB2(Зазначеного міста немає в списку)
 - Очікуваний результат - вивід повідомлення “Undefined city” та активне поле для вводу інформації
 - Результат роботи програми :

```
Choose the city :
['Prague' 'Paris' 'Amsterdam' 'Vienna' 'Berlin' 'London' 'Grindelwald'
 'Strasbourg' 'Florence' 'Salzburg' 'Rome' 'Dublin' 'Copenhagen'
 'Budapest' 'Oslo' 'Barcelona' 'Venice' 'Cappadocia' 'Stockholm' 'Munich']
Kiev
Undefined city !
```

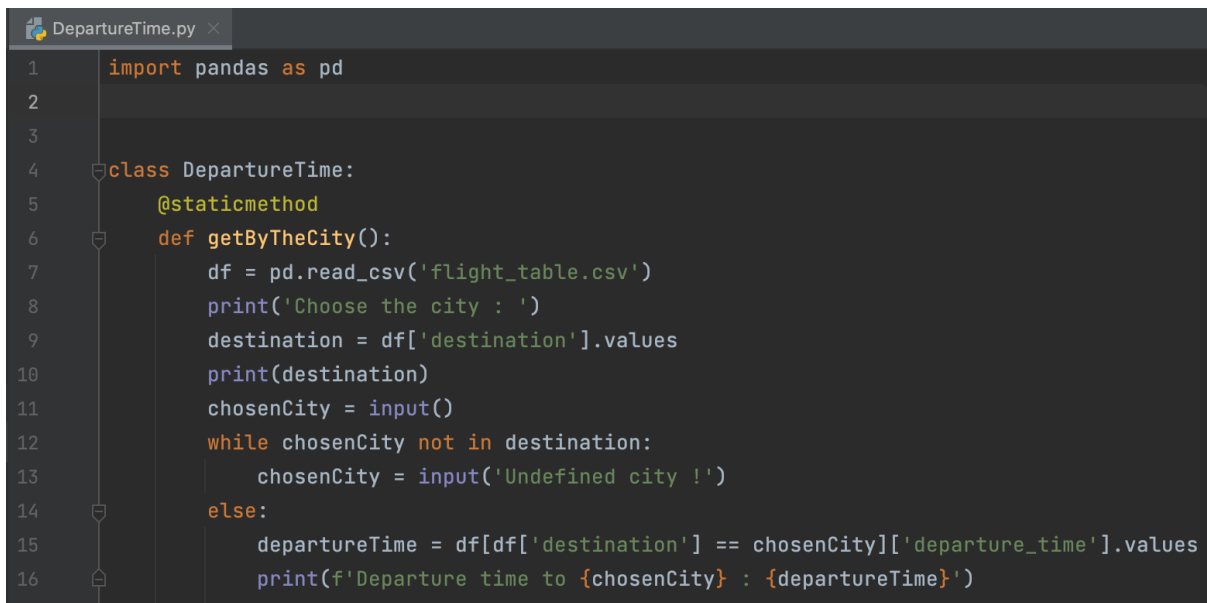
- TB3(В полі визначення міста зазначено числове значення)
 - Очікуваний результат - вивід повідомлення “Undefined city” та активне поле для вводу інформації
 - Результат роботи програми :

```
Choose the city :
['Prague' 'Paris' 'Amsterdam' 'Vienna' 'Berlin' 'London' 'Grindelwald'
 'Strasbourg' 'Florence' 'Salzburg' 'Rome' 'Dublin' 'Copenhagen'
 'Budapest' 'Oslo' 'Barcelona' 'Venice' 'Cappadocia' 'Stockholm' 'Munich']
!
Undefined city !
```

- ТВ4(Зазначене місто є в переліку , але назва введена з маленької літери)
 - Очікуваний результат - вивід повідомлення “Undefined city” та активне поле для вводу інформації
 - Результат роботи програми :

```
Choose the city :
['Prague' 'Paris' 'Amsterdam' 'Vienna' 'Berlin' 'London' 'Grindelwald'
 'Strasbourg' 'Florence' 'Salzburg' 'Rome' 'Dublin' 'Copenhagen'
 'Budapest' 'Oslo' 'Barcelona' 'Venice' 'Cappadocia' 'Stockholm' 'Munich']
prague
Undefined city !
```

AvailablePlaces.py



```
1 import pandas as pd
2
3
4 class DepartureTime:
5     @staticmethod
6     def getByTheCity():
7         df = pd.read_csv('flight_table.csv')
8         print('Choose the city : ')
9         destination = df['destination'].values
10        print(destination)
11        chosenCity = input()
12        while chosenCity not in destination:
13            chosenCity = input('Undefined city !')
14        else:
15            departureTime = df[df['destination'] == chosenCity]['departure_time'].values
16            print(f'Departure time to {chosenCity} : {departureTime}')
```

AvailablePlaces.py

- ТВ1 (Тест на правильність роботи функції для випадку, коли вхідні дані відсутні.)
 - Очікуваний результат - вивід повідомлення “Undefined city” та активне поле для вводу інформації
 - Результат роботи програми :

```
Choose the city :
['Prague' 'Paris' 'Amsterdam' 'Vienna' 'Berlin' 'London' 'Grindelwald'
 'Strasbourg' 'Florence' 'Salzburg' 'Rome' 'Dublin' 'Copenhagen'
 'Budapest' 'Oslo' 'Barcelona' 'Venice' 'Cappadocia' 'Stockholm' 'Munich']

Undefined city !
```

- TB2(Зазначеного міста немає в списку)
 - Очікуваний результат - вивід повідомлення “Undefined city” та активне поле для вводу інформації
 - Результат роботи програми :

```
Choose the city :
['Prague' 'Paris' 'Amsterdam' 'Vienna' 'Berlin' 'London' 'Grindelwald'
 'Strasbourg' 'Florence' 'Salzburg' 'Rome' 'Dublin' 'Copenhagen'
 'Budapest' 'Oslo' 'Barcelona' 'Venice' 'Cappadocia' 'Stockholm' 'Munich']

Kiev
Undefined city !
```

- TB3(В полі визначення міста зазначено числове значення)
 - Очікуваний результат - вивід повідомлення “Undefined city” та активне поле для вводу інформації
 - Результат роботи програми :

```
Choose the city :
['Prague' 'Paris' 'Amsterdam' 'Vienna' 'Berlin' 'London' 'Grindelwald'
 'Strasbourg' 'Florence' 'Salzburg' 'Rome' 'Dublin' 'Copenhagen'
 'Budapest' 'Oslo' 'Barcelona' 'Venice' 'Cappadocia' 'Stockholm' 'Munich']

1
Undefined city !
```

- TB4(Зазначене місто є в переліку , але назва введена з маленької літери)
 - Очікуваний результат - вивід повідомлення “Undefined city” та активне поле для вводу інформації
 - Результат роботи програми :

```
Choose the city :
['Prague' 'Paris' 'Amsterdam' 'Vienna' 'Berlin' 'London' 'Grindelwald'
 'Strasbourg' 'Florence' 'Salzburg' 'Rome' 'Dublin' 'Copenhagen'
 'Budapest' 'Oslo' 'Barcelona' 'Venice' 'Cappadocia' 'Stockholm' 'Munich']

prague
Undefined city !
```

Тестові варіанти для головного модуля :

- TB1(Тест на правильність роботи функції для випадку, коли вхідні дані відсутні.)
 - Очікуваний результат - вивід повідомлення “Undefined city” та активне поле для вводу інформації
 - Результат роботи програми :

```
Choose the action please . Available actions :  
1)Departure time(you should choose the city)  
2)Available places(you should choose the city)  
Action :  
Wrong action  
Action : |
```

- ТВ2(Введення “оманливого значення”)
 - Очікуваний результат - вивід повідомлення “Undefined city” та активне поле для вводу інформації
 - Результат роботи програми :

```
Choose the action please . Available actions :  
1)Departure time(you should choose the city)  
2)Available places(you should choose the city)  
Action :  
Wrong action  
Action : |
```

- ТВ3(Введення відповідних спецсимволів)
 - Очікуваний результат - вивід повідомлення “Undefined city” та активне поле для вводу інформації
 - Результат роботи програми :

```
Choose the action please . Available actions :  
1)Departure time(you should choose the city)  
2)Available places(you should choose the city)  
Action : !  
Wrong action  
Action : @  
Wrong action  
Action :
```

- ТВ4(Введення значення зі спецсимволами)
 - Очікуваний результат - вивід повідомлення “Undefined city” та активне поле для вводу інформації
 - Результат роботи програми :

```
Choose the action please . Available actions :  
1)Departure time(you should choose the city)  
2)Available places(you should choose the city)  
Action : "1"  
Wrong action  
Action : '1'  
Wrong action  
Action :
```

Висновок :

Для 11 лабораторної було виконане покрокове висхідне тестування . В результаті тестування за відповідними тест кейсами - можна сказати , що програма коректно працює з неправильними значеннями . Обробники умов написані таким чином, щоб унеможливити помилки . Досягнуто це шляхом того , що для вибору відповідної дії чи об'єкту порівнюються саме текстові значення , які було введено в консоль . Навіть для числових значень було використано цей принцип . Таким чином програма порівнює значення безпосередньо зі значеннями символів в ASCII таблиці .