

Мета: навчитись перехоплювати та опрацьовувати виняткові ситуації при складанні програм з використанням об'єктно-орієнтованого підходу на мові Python в інтерактивному режимі IDLE

Обладнання: ОС Windows, Microsoft Office2013, середовище програмування IDLE

Хід роботи

1 Використовуючи головне меню ОС Windows запустив інтерактивний режим інтерпретатора Python – середовище програмування IDLE

2 Створив в своєму каталогі папку ЛР20 для збереження наступних завдань

3 За допомогою команди File→NewFile (Ctrl+N) створив новий текстовий документ з від повідною назвою LP_20(test1) для тестування наступного коду програми

Задача 1: створити конструктор класу який визначає результат ділення двох цілих змінних; значення змінних x та у ввести з клавіатури в діалоговому режимі в основній частині програми та передати їх в клас Test1 через відповідних його екземпляр (протестуйте його двічі: для значень 12 і 6 та 5 і 0)

Код програми	===== RESTART: =====
<pre>class Test1: def __init__(self,x,y): print(int(x)/int(y)) a1 = Test1(input('x:'),input('y:')) a2 = Test1(input('x:'),input('y:'))</pre>	x:12 y:6 2.0 x:5 y:0
	Traceback (most recent call last): ...
	ZeroDivisionError: division by zero

Змн.	Арк.	№ докум.	Підпис	Дата
Розроб.	Мітленко Д.А.			
Перевір.	Саприкіна І.Г.			
Реценз.				
Н. Контр.	Саприкіна І.Г.			
Замерд.	Саприкіна І.Г.			

ДТЗЕ.121РПЗ00ТЕХ2000ЛР

**Розробка програм з
опрацювання
виняткових ситуацій**

Літ.	Арк.	Аркушів
	1	10
Група ПЗ-19-1/9		

Поясніть, чому виникла помилка при вводі значень 5 і 0. Тому, що ділення на нуль не існує.

Додайте до класу блок try опрацювання виняткової ситуації ZeroDivisionError для переходу лення помилки при діленні на нуль. Протестуйте програму з різними наборами тестів.

<i>Код програми</i>	<i>===== RESTART: =====</i>
class Test:	x-> 12
def __init__(self,a,b):	y-> 6
try:	rez= 2.0
re = a/b	програма завершена:)
except ZeroDivisionError:	x-> 5
print('Ділення на 0!!!')	y-> 0
else:	Ділення на 0!!!
print('rez=',re)	програма завершена:)
finally:	
print('програма завершена:)')	

4 За допомогою команди File→NewFile (Ctrl+N) створив новий текстовий документ з відповідною назвою LP_20(test2) для тестування наступного коду програми

Задача 2: на рахунок в банк покладено певну суму K, з банкомату необхідно зняти потрібну суму x. Створити клас який моделює наступні дії банкомата: якщо сума x не перевищує K, банкомат видає потрібну суму і повідомляє про остатчу коштів на рахунку; якщо ввести суму x яка перевищує допуск тимчасового ліміту по рахунку – банкомат не видає кошти і повідомляє про помилку вводу.

Змін.	Мітленко Д.А.		
Арк.	№ докум.	Підпис	Дата

ДТЗЕ.121РГ300ТЕХ2000ЛР

Арк.

2

Код програми

```
class Own_exception (Exception):
    def __init__(self,K,x):
        Exception. __init__(self)
        self.p1=K
        self.p2=x
        K=int(input("рахунок: "))
        try:
            x=int(input("яку суму знімаєте?: "))
            if x>K:
                raise Own_exception (K,x)
            except Own_exception as rez:
                print("Таку суму зняти не
можна!!!") print("на рахунку
",K,"грн")
            else:
                s=K-x
                print("знято: ",x, "остача на рахунку",s)
        finally:
            print("програма завершена")
```

===== *RESTART:* =====

рахунок: 5000
яку суму знімаєте?: 1500
знято: 1500, остатча на
рахунку 3500
рахунок: 5000
яку суму знімаєте?: 6450
Таку суму зняти не можна!!!
на рахунку 5000 грн

Для реалізації програми використати наслідування класу вбудованих винятків
Exception

Змін.	Арк.	Мітленко Д.А.		

ДТЗЕ.121РГ300ТЕХ2000ЛР

Арк.

3

<p><i>Код програми</i></p> <pre> class Own_exception (Exception): def __init__(self,K,x): Exception. __init__(self) self.p1=K self.p2=x K=int(input("рахунок: ")) try: x=int(input("яку суму знімаєте?: ")) if x>K: raise Own_exception (K,x) except Own_exception as rez: print("Таку суму зняти не можна!!!!") print("на рахунку ",K,"грн") else: s=K-x print("знято: ",x, "остача на рахунку",s) finally: print("програма завершена") </pre>	<p>===== <i>RESTART:</i> =====</p> <p>рахунок: 5000</p> <p>яку суму знімаєте?: 1250</p> <p>знято: 1250 остатча на рахунку 3750</p> <p>програма завершена</p>
--	--

Опис компонентів програми:

class Own_exception (Exception) – створення класу, який наслідує помилку

raise Own_exception (K,x) – викликати помилку

except Own_exception as rez – перехват помилки

5 За допомогою команди File→NewFile (Ctrl+N) створив нові текстові документи з відповідними назвами ЛР_20(№ завдання) для розробки наступних кодів програми.

Змін.	Мітленко Д.А.		
Арк.	№ докум.	Підпис	Дата

ДТЗЕ.121РГ300ТЕХ2000ЛР

Арк.

6 Виконав індивідуальні завдання виданого варіанту (варіант 1)

6.1 Розробіть програму генерування двох чисел в заданому діапазоні та їхнього додавання, якщо перше число більше другого, інакше – їх множення. Зробіть синтаксичну помилку в записі оператора виводу на екран та перехопіть її через виняток класу `SyntaxError` для нормального завершення програми

Код программы:

```
from random import randint
```

try:

```
a = randint(0,100)  
b = randint(0,100)  
if a>b:  
    eval("print('a + b =',a+b))")
```

else:

Chap. 1. — Sect. 4. — Elements.

```
print("Помилка виконання програми!")
```

Результат зображеній на скріншоті 1



Python 3.7.9 Shell

File Edit Shell Debug Options Window Help

Python 3.7.9 (tags/v3.7.9:13c94747c7, Aug 17 2020, 18:58:18) [MSC v.1900 64 bit (AMD64)] on win32

Type "help", "copyright", "credits" or "license()" for more information.

>>>

===== RESTART: E:\Archive\+LP\tech\лр20\20_1.py =====

Помилка виконання програми

>>> |

Скріншот 1

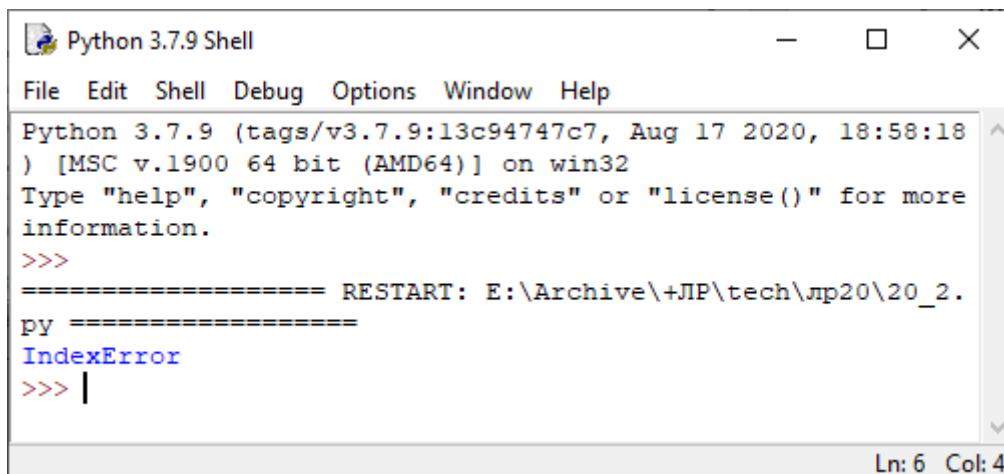
6.2 Розробіть програму для циклічного створення списку з 5 елементів і звернення до 8 елемента. Використайте виняток IndexError для перехоплення відповідної винятку, опрацювання та нормального завершення програми. Додайте ще один виняток для опрацювання некоректного значення.

		<i>Мітленко Д.А.</i>				Арк.
Змін.	Арк.	№ докум.	Підпис	Дата	ДТЗЕ.121РП300TEX2000ЛР	5

Код програми:

```
from random import random
arr = [random() for i in range(5)]
try:
    print(arr[8])
except IndexError:
    print('IndexError')
except ValueError:
    print('ValueError')
```

Результат зображеній на скріншоті 2



The screenshot shows the Python 3.7.9 Shell window. The title bar says "Python 3.7.9 Shell". The menu bar includes File, Edit, Shell, Debug, Options, Window, and Help. The main window displays the Python interpreter's prompt and some code. It starts with the Python version information: "Python 3.7.9 (tags/v3.7.9:13c94747c7, Aug 17 2020, 18:58:18) [MSC v.1900 64 bit (AMD64)] on win32". It then shows the help message: "Type "help", "copyright", "credits" or "license()" for more information.". The user enters ">>>" followed by "py" which triggers a restart. The next line shows "IndexError" in blue, indicating it is an error. The bottom status bar shows "Ln: 6 Col: 4".

Скріншот 2

6.3 Переробіть програму 1 завдання таким чином, щоб одна із змінних була не визначеною. Використайте виняток `NameError` для перехоплення, опрацювання цієї події і нормального завершення програми.

Код програми:

```
from random import randint
try:
    a = randint(0,100)
    b = randint(0,100)
    if a>b:
        print('a + b =',a+b)
    else:
```

Змін.	Арк.	Мітленко Д.А.		

```

print('a * b =',a*b)
except NameError:
    print("Помилка виконання програми")

```

Результат зображенний на скріншоті 3

The screenshot shows a Python 3.7.9 Shell window. The title bar says "Python 3.7.9 Shell". The menu bar includes File, Edit, Shell, Debug, Options, Window, Help. The main window displays the following text:

```

Python 3.7.9 (tags/v3.7.9:13c94747c7, Aug 17 2020
, 18:58:18) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()"
" for more information.

>>>
===== RESTART: E:\Archive\+LP\tech\
лр20\20_3.py =====
Помилка виконання програми
>>> |

```

Ln: 6 Col: 4

Скріншот 3

6.4 Розробити тестову програму яка пропонує користувачу відповісти на тестові питання. Формування відповіді у форматі «правильно» і «не правильно» опрацювати за допомогою відповідного винятку через наслідування класу Exception. У будь-якому випадку програма повинна коректно працювати для формування відповідей на 5 запитань і правильно завершувати свою роботу.

Код програми:

```

test = {
    0:{'t':'2 + 2 = 4?', 'a':True},
    1:{'t':'2 + 2 * 2 = 8?', 'a':False}
}

```

Змін.	Мітленко Д.А.		
Арк.	№ докум.	Підпис	Дата

```

    },
2:{'t':'47 / 22 = 3?',
     'a':False
},
3:{'t':'99 / 11 = 9',
     'a':True
},
4:{'t':'22 + 7 - (18 * 22) = -367',
     'a':True
},
i = 0
while True:
    try:
        print('\nПитання:',test[i]['t'])
        an = input('(правильно/не правильно) >')
        if (an == 'правильно' and test[i]['a'] == True) or (an == 'не правильно' and test[i]['a'] == False):
            i += 1
            print('Вірно!')
        elif not an in ('правильно','не правильно'):
            raise Exception('Введіть "правильно" або "не правильно"!')
        else:
            raise Exception('Не вірно! Спробуйте ще раз.')
    except Exception as e:
        print(e)
        if i == 5: break

```

		<i>Мітленко Д.А.</i>		
Змін.	Арк.	№ докум.	Підпис	Дата

Результат зображеній на скріншоті 4

The screenshot shows a Python 3.7.9 Shell window. The code runs a series of math questions. For each question, it asks for the answer and whether it is correct ('правильно') or incorrect ('не правильно'). It also asks for a response ('Вірно!' or 'Ні'). The window title is 'Python 3.7.9 Shell'. The status bar at the bottom right shows 'Ln: 29 Col: 4'.

```
Python 3.7.9 (tags/v3.7.9:13c94747c7, Aug 17 2020, 18:58:18) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
=====
RESTART: E:\Archive\+ЛР\tech\лр20\20_4.py =====

Питання: 2 + 2 = 4?
(правильно/не правильно) >правильно
Вірно!

Питання: 2 + 2 * 2 = 8?
(правильно/не правильно) >не правильно
Введіть "правильно" або "не правильно"!

Питання: 2 + 2 * 2 = 8?
(правильно/не правильно) >не правильно
Вірно!

Питання: 47 / 22 = 3?
(правильно/не правильно) >не правильно
Вірно!

Питання: 99 / 11 = 9
(правильно/не правильно) >правильно
Вірно!

Питання: 22 + 7 - (18 * 22) = -367
(правильно/не правильно) >правильно
Вірно!
>>> |
```

Скріншот 4

Відповіді на контрольні запитання:

1 Що називають винятком в програмі?

Виняток – це подія, яка може виникнути під час виконання програми та змінити подальший хід її виконання

2 Які стандартні винятки можуть виникати в програмі?

В Python виділяють два різних види помилок: синтаксичні помилки і виключення

3 Що повідомляється користувачу, коли виникає виняток?

Назва файлу, рядок, тип помилки

4 Що таке парсинг програми?

Виконання дій записаних у фалі певною мовою програмування.

	Mітленко Д.А.				Арк.
Змін.	Арк.	№ докум.	Підпис	Дата	ДТЗЕ.121РГ300TEX2000ЛР

5 За допомогою якої інструкції перехоплюються винятки?

Try – except

6 Яку структуру має блок try?

Try:

Diya

Except Exception as e:

Diya pri pomilki

7 Для чого призначена структура raise?

Для створення виключень вручну

Висновок: я навчився перехоплювати та опрацьовувати виняткові ситуації при складанні програм з використанням об'єктно-орієнтованого підходу на мові Python в інтерактивному режимі IDLE

Змін.	Арк.	Мітленко Д.А.		

ДТЗЕ.121РП300ТЕХ2000ЛР

Арк.

10