python

Programming Essentials

Day 7

Speaker: Vincent M. Musa
Content by: Cyrel P. Nicolas

The + operator can be used to concatenate two or more tuples together.

python
Programming Essentials
Day 7

Random Module

Python has a built-in module that you can use to make random numbers.

randint()

If we wanted a random integer, we can use the randint() function Randint accepts two parameters: a lowest and a highest number.

```
import random
print(random.randint(0, 9))
```

Multiple randint() Method Call

```
import random
for i in range(5):
    print(random.randint(1, 10))
```

Randomly Generated Output

```
3
5
1
3
9
```


Speaker: Vincent M. Musa
Content by: Cyrel P. Nicolas

The + operator can be used to concatenate two or more tuples together.

python
Programming Essentials
Day 7

Day 7 Act 1

1. Create a Name Generator App

2. Create 3 lists for first name, middle name, and last name with 10 items per list

3. The application will ask the user to generate a new name.

4. If yes, use a random number between 0 - 9 to randomly select items in the lists

5. Display the generated name “Congratulations! Your new name is _____”

6. If No, display the word “Thank you! ” and display all the names that user generated.

The + operator can be used to concatenate two or more tuples together.

python
Programming Essentials
Day 7

What are Exceptions?

A Python program terminates as soon as it encounters an error. In Python, an error can be a syntax error or an exception.

- Events or errors that disrupt the normal flow of execution of a program.
- It prevents the program from reaching a normal end.
- These events or errors usually occur during runtime.

Example of exceptions:

- Division by zero
- Invalid input
- File not found

Kinds of Exceptions

Checked Exception

- All exceptions, except for Runtime Exception, are checked exceptions.

- Exceptions are “checked” because they are subject to the Catch or Specify Requirement. Otherwise, the program code will not compile.

- Checked exceptions are errors that the program can deal with.

Kinds of Exceptions

Errors Exception

- Errors are generally beyond the control of the program. These are situations that cannot be anticipated and for which the program cannot recover from.

- Example:
  - Unreadable file
  - Hardware malfunction

- Errors are not subject to the Catch or Specify requirement, and are often referred to as unchecked exceptions.

The + operator can be used to concatenate two or more tuples together.

Kinds of Exceptions

Runtime Exception

- Like errors, Runtime Exceptions are not subject to the Catch or Specify Requirement and are, also, unchecked exceptions.

- These exceptions are the result of programming flaws such as:
  - dividing by zero,
  - using null pointers or references,
  - or going beyond an array's boundaries.

The + operator can be used to concatenate two or more tuples together.

python
Programming Essentials
Day 7

Handling Exceptions

Runtime Exception

```
Try:
        pass
Except ExceptionName:
        pass
Else:
        pass
Finally:
        pass
```

The + operator can be used to concatenate two or more tuples together.

Handling Exceptions

python
Programming Essentials
Day 7

```
try:
    result = 100 / 0
except ZeroDivisionError:
    print("Division by zero is error !!")
except:
    print("Wrong input")
else:
    print("No exceptions.", "Display the result is: ", result)

finally:
    print("This will execute no matter what")
```

Output:

```
Division by zero is error !!
This will execute no matter what
```


Cyrel P. Nicolas

Handling Exceptions

```
try:
    result = 100 / a
except ZeroDivisionError:
    print("Division by zero is error !!")
except:
    print("Wrong input")
else:
    print("No exceptions.", "Display the result is: ", result)

finally:
    print("This will execute no matter what")
```

Output:

```
Wrong input
This will execute no matter what
```

The + operator can be used to concatenate two or more tuples together.

Handling Exceptions

python
Programming Essentials
Day 7

```
try:
    result = 100 / 2
except ZeroDivisionError:
    print("Division by zero is error !!")
except:
    print("Wrong input")
else:
    print("No exceptions.", "Display the result is: ", result)

finally:
    print("This will execute no matter what")
```

Output:

```
No exceptions. Display the result is:  50.0
This will execute no matter what
```


Cyrel P. Nicolas

The + operator can be used to concatenate two or more tuples together.

Try Block

The try block lets you test a block of code for errors.

```
try:
    file = open("filename.txt")
```

Except Block

The Except block will be executed if the try block raises an error.

```
try:
    file = open("filename.txt")
except FileNotFoundError as fileError:
    print(fileError)
```

```
[Errno 2] No such file or directory: 'filename.txt'
```

Else Block

You can use the else keyword to define a block of code to be executed if no errors were raised:

```
try:
    file = open("info.txt")
except FileNotFoundError as fileError:
    print(fileError)
else:
    print("File is Found!")
```

The + operator can be used to concatenate two or more tuples together.

Finally Block

The finally block, if specified, will be executed regardless if the try block raises an error or not.

```
try:
    file = open("info.txt")
except FileNotFoundError as fileError:
    print(fileError)
else:
    print("File is Found!")
finally:
    file.close()
```

The + operator can be used to concatenate two or more tuples together.

python
Programming Essentials
Day 7

Day 7 Act 2

1. Create a calculator app
2. The user will choose between the 4 math operations (Add, Subtract, Multiply and Divide)
3. The application will ask for 2 numbers
4. Display the result
5. The application will ask again if the user wants to try again
6. Use the appropriate Exception (ex: Invalid input such as text and zero division)

Cyrel P. Nicolas