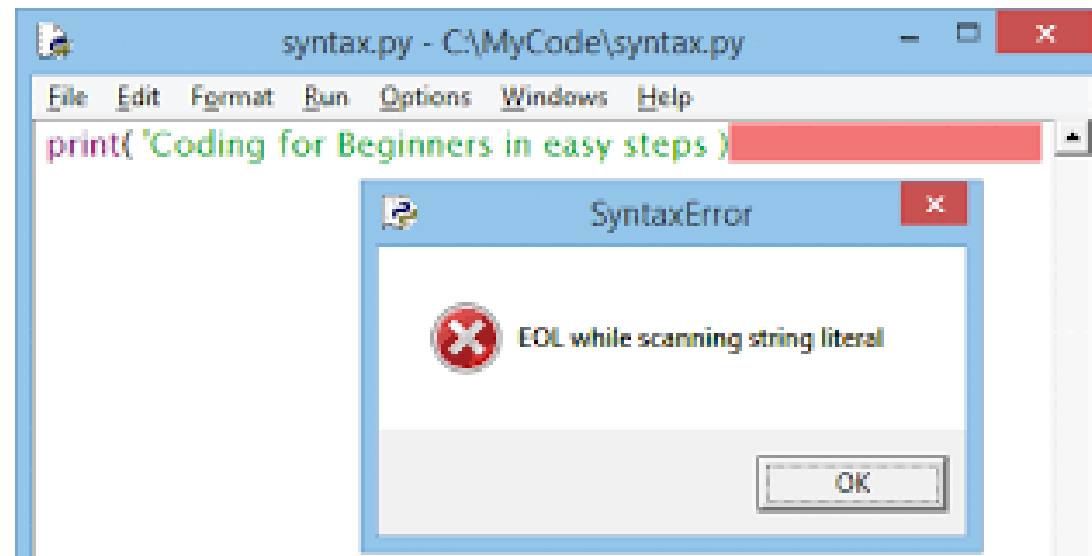


Task

Create a table or mind map about programming errors
It MUST include a definition of the error and an example

- logical
- syntactical (syntax error)
- run-time/execution
- linking
- rounding
- truncation



To develop knowledge by
Describing different
programming errors

Use the code to create the programming errors. SCREENSHOT your code and result into the worksheet

1. Syntax error

Write definition then create a syntax error in Python

```
num1 = 456  
  
if num1 == 2  
    print("Wrong")
```

2. Logical error

Write definition then create a runtime error using the code below:

```
age = 55  
  
if age <18:  
    print("You are an adult")  
else:  
    print("You are a child")
```

3. Runtime error

Write definition then create a runtime error using this code

```
File Edit Format Run Options Window Help  
temp = 0  
  
answer = 1/temp # 1 divided by ?  
print(answer)
```

Spirit:

- ✓ Self management
- ✓ Independence

4. Linking error

Write definition then create a linking error using the code below:

```
#amend this code to produce a linking error  
import random  
  
number = random.randint(1,3)  
print(number)
```

To develop knowledge by

Describing different programming errors

To secure understanding by

applying theory to create programming errors.

Excellence

S.P.I.R.I.T

- ✓ Self management
- ✓ Innovation

1. Write definitions of the following more complex errors

Errors:

- Rounding
- Truncation

```
>>> round(3.56, 1) ← Correct
3.6
>>>
>>> round(3.56) ← Incorrect
4
>>> |
```

Create own examples of errors in Python and screenshot

Hints: investigate how to use the round and trunc functions in python to do this task

Rounding
round()

Truncating:

```
import math

num = math.trunc()
```

To achieve excellence by
Investigating more complex errors and
creating examples

Excellence

- Complete the exams question in your book.

Errors often occur in programming.

Describe 3 types of common errors [6]

5. Computer programs require *translation* to execute.

(a) *Compilers* and *interpreters* translate high level programming languages into machine code. Describe the main differences between a compiler and an interpreter. [4]

.....