1. Write Python program to demonstrate the following:

1. SyntaxError
2. TypeError
3. IndexError
4. ValueError
5. ZeroDivisionError
6. fileNotFound

```
# SyntaxError
print "Hello, World!"


# TypeError
result = "5" + 7


# IndexError
my_list = [1, 2, 3]
print(my_list[7])


# ValueError
num = int("abc")


# ZeroDivisionError
result = 5 / 0


# FileNotFoundError
with open("nonexistent_file.txt") as file:
    content = file.read()
```

```
  File "C:\Users\Vandith\AppData\Local\Temp\ipykernel_5552\3952587861.py", line 2
    print "Hello, World!"  # Uncomment this line to see SyntaxError
          ^
SyntaxError: Missing parentheses in call to 'print'. Did you mean print("Hello, World!"  # Uncomment this line to see SyntaxErr
or)?
```

2. Write Python program to raise user defined exception

```
class AgeException(Exception):
    pass


age=int(input("Enter your age"))


def validate_age(voter_age):
    if voter_age<18:
        print("You can't vote")
    if voter_age>=18:
        print("You can vote")
    if voter_age < 0:
        raise AgeException("Age cannot be negative.")


try:
    validate_age(age)
except AgeException as ae:
    print("Custom Exception occurred:", ae)
```

```
Enter your age-10
You can't vote
Custom Exception occurred: Age cannot be negative.
```

3. Write Python program to demonstrate the use of try, except and finally block

```
try:
    num = int(input("Enter a number: "))
    result = 10 / num
    print("Result:", result)
except ZeroDivisionError:
    print("Cannot divide by zero.")
finally:
```

```
    print("This block always executes.")
```

```
Enter a number: 0
Cannot divide by zero.
This block always executes.
```

4. Write Python program to demonstrate default except block

```
try:
    num = int(input("Enter a number: "))
    result = 10 / num
    print("Result:", result)
except (ValueError, ZeroDivisionError):
    print("Invalid input or division by zero.")
```

```
Enter a number: Vk
Invalid input or division by zero.
```

5. Write Python program to handle multiple exceptions in single except block

```
try:
    num = int(input("Enter a number: "))
    result = 10 / num
    print("Result:", result)
except (ValueError, ZeroDivisionError):
    print("Invalid input or division by zero.")
```

```
Enter a number: Vk
Invalid input or division by zero.
```

6. Write a program to read the contents of file and perform following operations

a) display number of words

b) display number of characters

c) display number of vowels

d) display number of lines

e) reverse each word and display it

```
def count_words(file_content):
    return len(file_content.split())

def count_characters(file_content):
    return len(file_content)

def count_vowels(file_content):
    vowels = "aeiouAEIOU"
    return sum(1 for char in file_content if char in vowels)

def count_lines(file_content):
    return file_content.count('\n') + 1

def reverse_words(file_content):
    words = file_content.split()
    reversed_words = [word[::-1] for word in words]
    return ' '.join(reversed_words)

try:
    with open("sample.txt", "r") as file:
        content = file.read()
        print("Number of words:", count_words(content))
        print("Number of characters:", count_characters(content))
        print("Number of vowels:", count_vowels(content))
        print("Number of lines:", count_lines(content))
        print("Reversed words:")
        print(reverse_words(content))
except FileNotFoundError:
```

```
print("File not found.")
```

```
Number of words: 9
Number of characters: 41
Number of vowels: 12
Number of lines: 3
Reversed words:
iiH yM eman si htidnaV I ma gnihctaw LPI
```