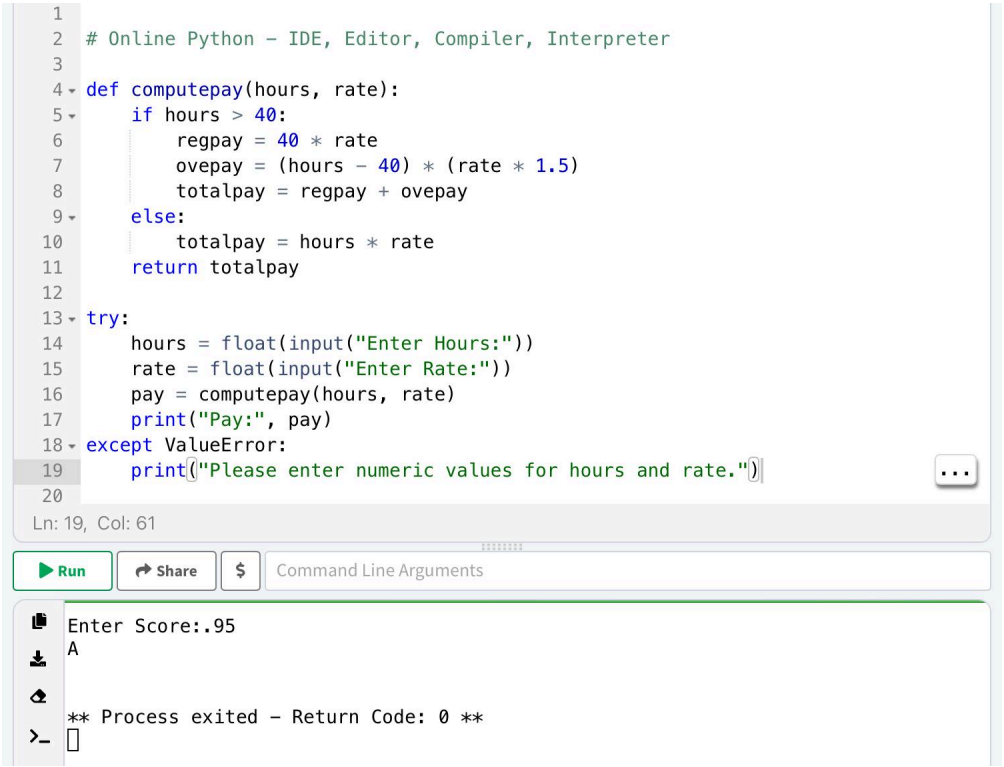
DE JESUS, CLAIRE JUSTINE G
IE 22S1 CPE 310 EXERCISE 1.2

Part 1: Functions

Exercise 1: (d) b and c are both true
Exercise 2: (d) ABCZapABC

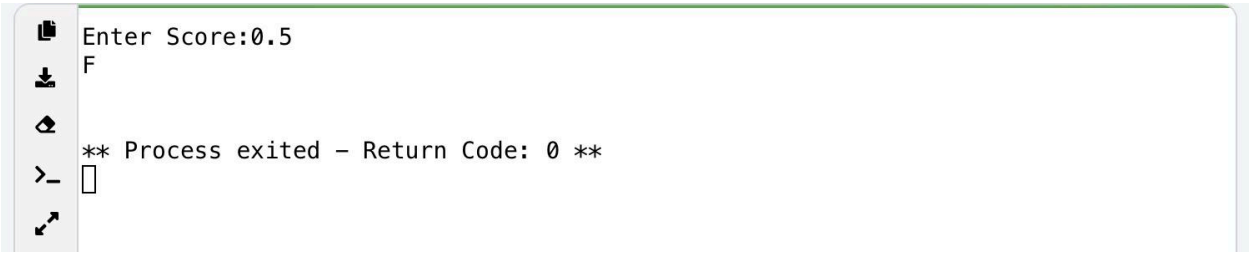
Exercise 3:

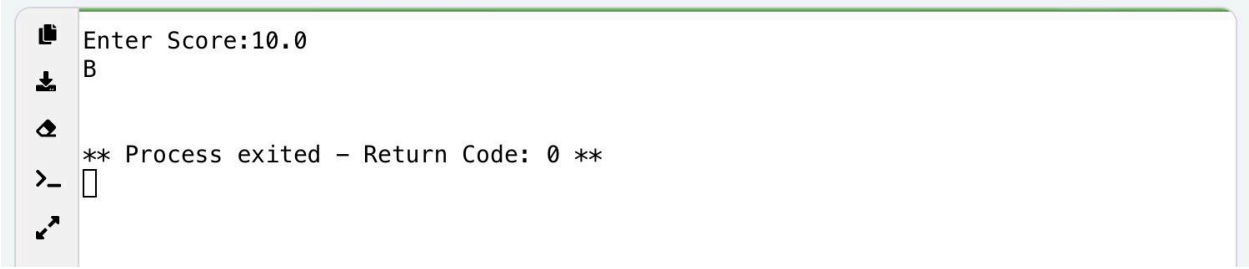
```
def computepay(hours, rate):
    if hours > 40:
        regpay = 40 * rate
        ovepay = (hours - 40) * (rate * 1.5)
        totalpay = regpay + ovepay
    else:
        totalpay = hours * rate
    return totalpay

try:
    hours = float(input("Enter Hours:"))
    rate = float(input("Enter Rate:"))
    pay = computepay(hours, rate)
    print("Pay:", pay)
except ValueError:
    print("Please enter numeric values for hours and rate.")
```

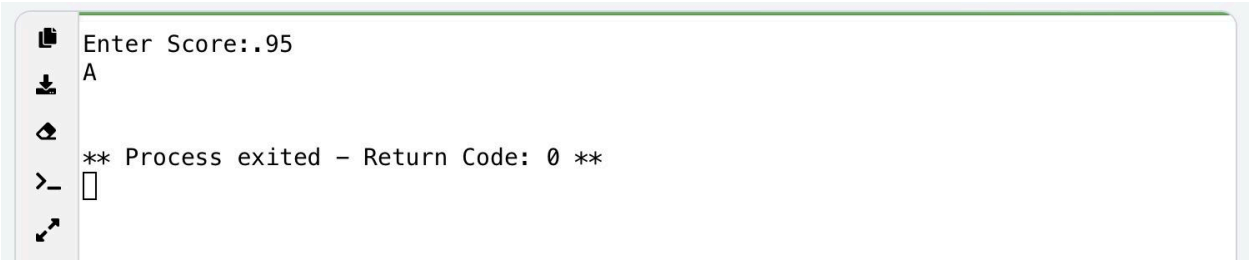


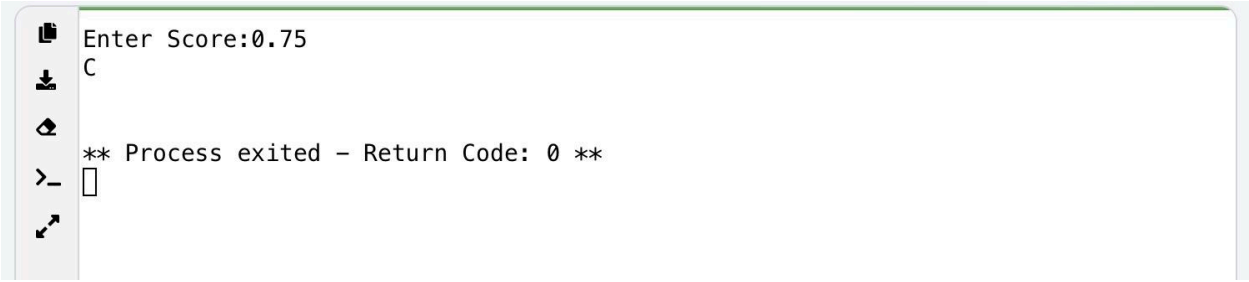
Exercise 4:

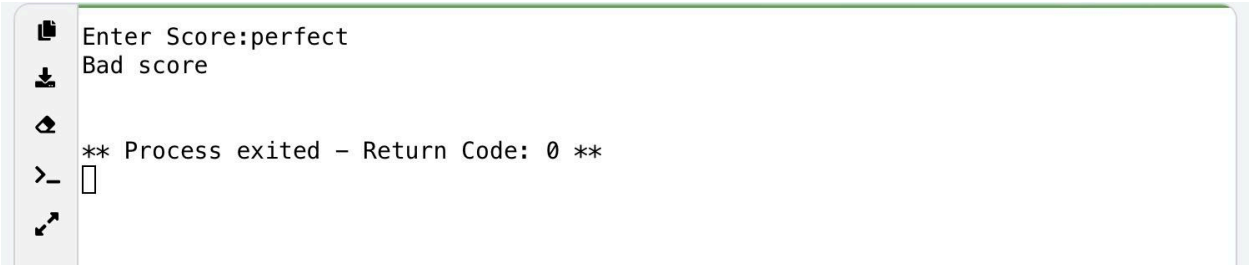
```
def computegrade(score):
    if score >= 0.9 and score <= 1.0:
        return "A"
    elif score >= 0.8:
        return "B"
    elif score >= 0.7:
        return "C"
    elif score >= 0.6:
        return "D"
    elif score >= 0.0:
        return "F"
    else:
        return "Bad score"
try:
    entscore = input("Enter Score:")
    score = float(entscore)

    grade = computegrade(score)
    print(grade)
except ValueError:
    print("Bad score")
```











Part 2: Loops and Iterations

Exercise 1:

```
total = 0
count = 0

while True:
    entnum = input("Enter a number:")
    if entnum.lower() == "done":
        break
    try:
        number = int(entnum)
        total += number
        count += 1
    except ValueError:
        print("Invalid Input")

if count > 0:
    average = total/count
    print(total, count, average)
```

main.py Untitled2.py Untitled3.py Untitled4.py +

```
1  total = 0
2  count = 0
3
4  while True:
5      entnum = input("Enter a number:")
6      if entnum.lower() == "done":
7          break
8      try:
9          number = int(entnum)
10         total += number
11         count += 1
12     except ValueError:
13         print("Invalid Input")
14
15 if count > 0:
16     average = total/count
17     print(total, count, average)
```

Ln: 17, Col: 32

Run Share $ Command Line Arguments

```
Enter a number:4
Enter a number:5
Enter a number:bad data
Invalid Input
Enter a number:7
Enter a number:done
16 3 5.333333333333333


** Process exited - Return Code: 0 **
```

Exercise 2:

```
numbers = []

while True:
    entnum = input("Enter a number:")
    if entnum.lower() == "done":
        break
    try:
        number = int(entnum)
        numbers.append(number)
    except ValueError:
        print("Invalid input")

if numbers:
```

```
    print(f"Maximum = {max(numbers)}, Minimum = {min(numbers)}")
else:
    print("No numbers were enetered.")
```

main.py | Untitled2.py | Untitled4.py | Untitled5.py | Untitled7.py | +

```
1  numbers = []
2
3  while True:
4      entnum = input("Enter a number:")
5      if entnum.lower() == "done":
6          break
7      try:
8          number = int(entnum)
9          numbers.append(number)
10     except ValueError:
11         print("Invalid input")
12
13 if numbers:
14     print(f"Maximum = {max(numbers)}, Minimum = {min(numbers)}")
15 else:
16     print("No numbers were enetered.")
```

Ln: 4, Col: 9

Run | Share | $ | Command Line Arguments

```
Enter a number:2
Enter a number:3
Enter a number:4
Enter a number:5
Enter a number:done
Maximum = 5, Minimum = 1


** Process exited - Return Code: 0 **
```