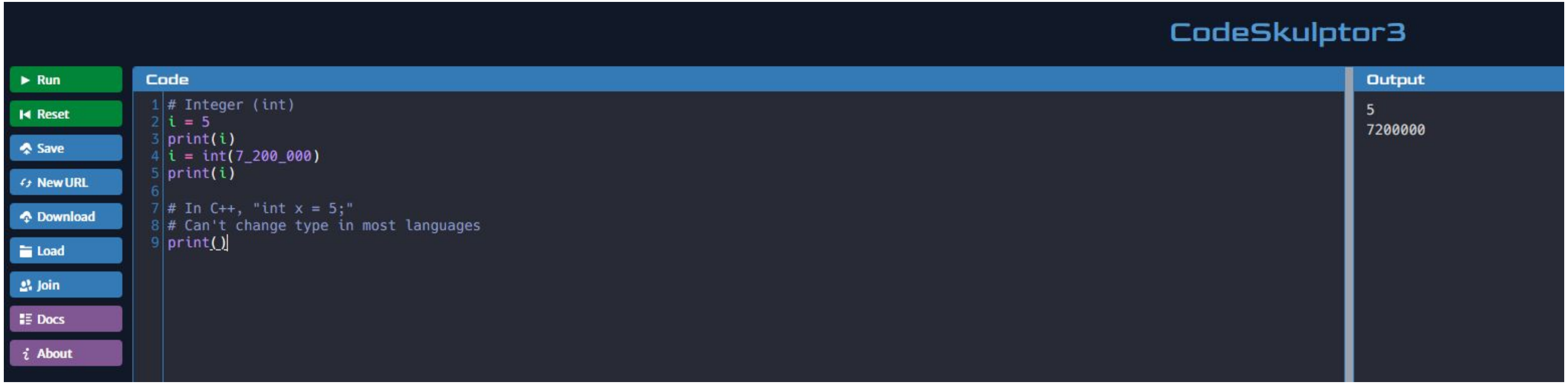
Programming Club Meeting 2 Slides

Meeting 1 Review

About the Club

- I am James a senior here
- I am Jon and I am a sophomore here
- The main goal of the club is to get better at programming
- There will be competitions, you don’t have to join but are welcome to
- Your experience in programming isn’t important, the overall goal is just to get better

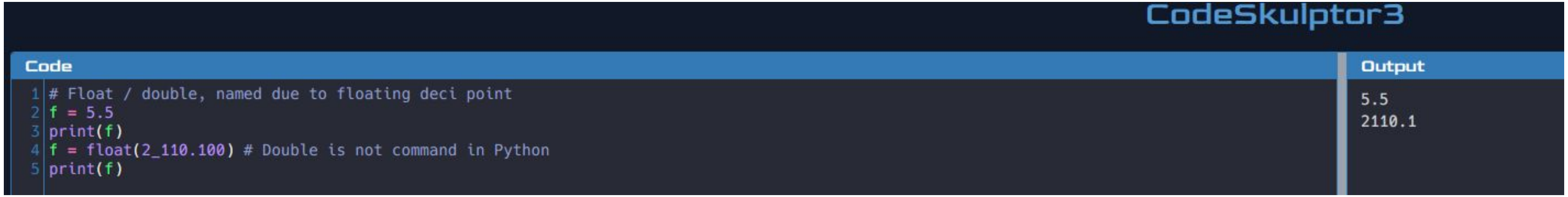
Data Types P1



```
# Integer (int)
i = 5
print(i)
i = int(7_200_000)
print(i)

# In C++, "int x = 5;"
# Can't change type in most languages
print()
```

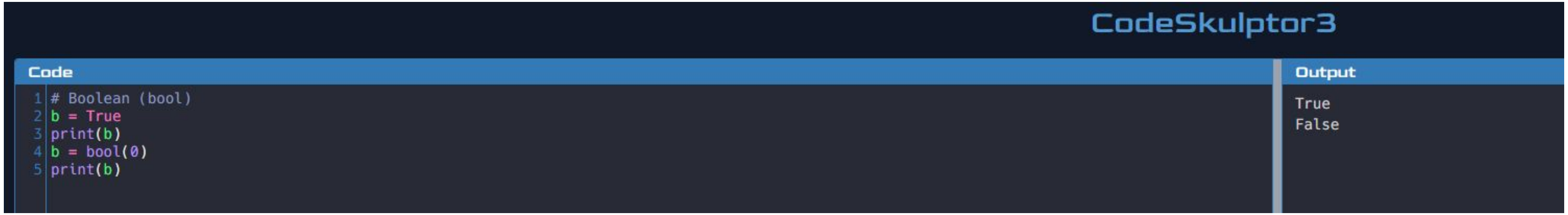
```
5
7200000
```



```
# Float / double, named due to floating deci point
f = 5.5
print(f)
f = float(2_110.100) # Double is not command in Python
print(f)
```

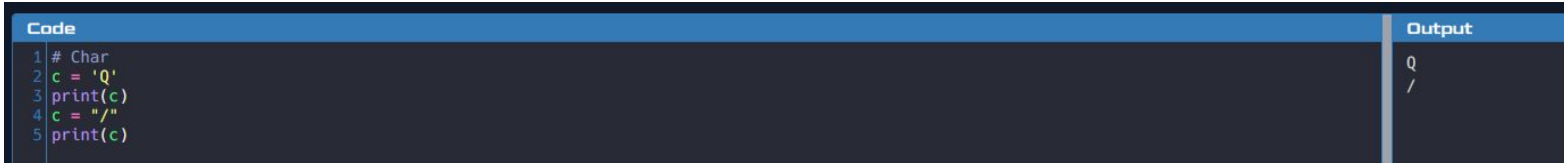
```
5.5
2110.1
```

Data Types P2



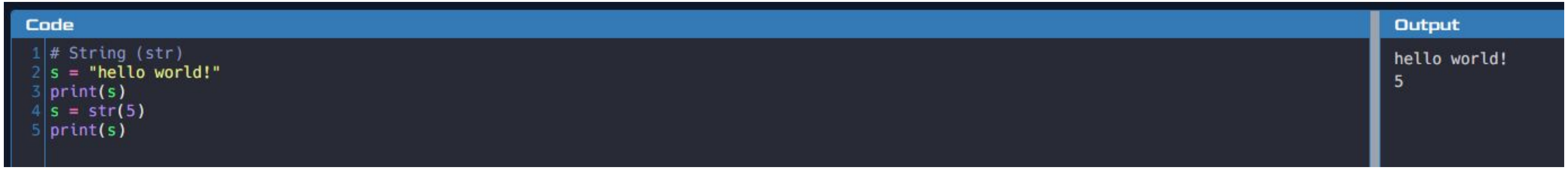
```
# Boolean (bool)
b = True
print(b)
b = bool(0)
print(b)
```

```
True
False
```



```
# Char
c = 'Q'
print(c)
c = "/"
print(c)
```

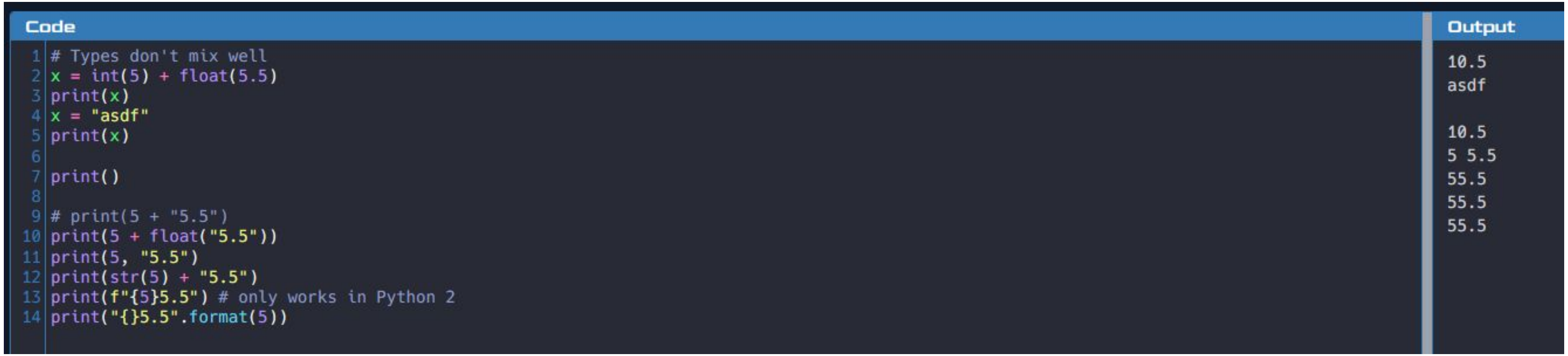
```
Q
/
```



```
# String (str)
s = "hello world!"
print(s)
s = str(5)
print(s)
```

```
hello world!
5
```

Data Types P3



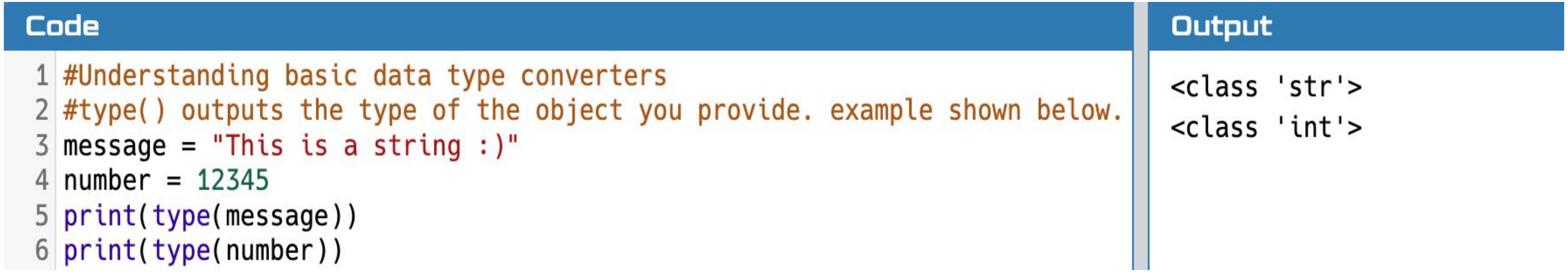
```python
# Types don't mix well
x = int(5) + float(5.5)
print(x)
x = "asdf"
print(x)

print()

# print(5 + "5.5")
print(5 + float("5.5"))
print(5, "5.5")
print(str(5) + "5.5")
print(f"{5}5.5") # only works in Python 2
print("{}5.5".format(5))
```

```
10.5
asdf

10.5
5 5.5
55.5
55.5
55.5
```



```python
#Understanding basic data type converters
#type( ) outputs the type of the object you provide. example shown below.
message = "This is a string :)"
number = 12345
print(type(message))
print(type(number))
```

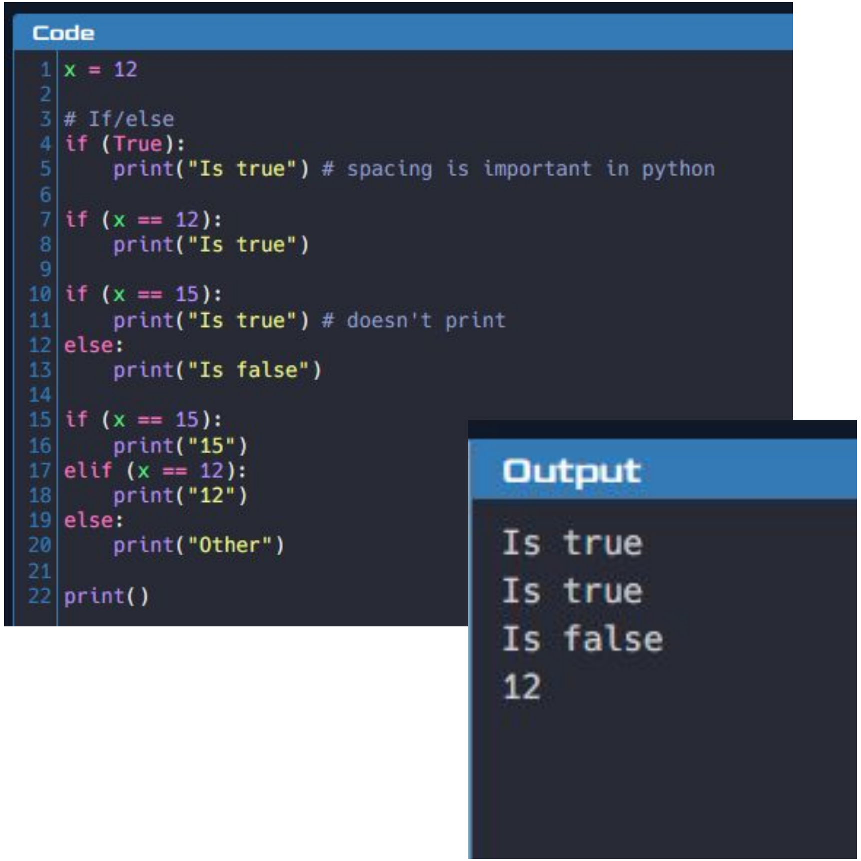
```
<class 'str'>
<class 'int'>
```

Also Covered:

- Basic Arithmetic: +, -, *, /
- “Weird” Arithmetic: %, //, **, sqrt
- Assignment (Variable Creation): =, +=, -=, *=, /=
- Comparison: ==, !=, >, <
- Combined Comparison: and, or, not

Selection (If/Else)

- Covered last meeting but not in depth
- If the condition is true, then the code inside will run
- Allows much more flexibility in programs
- Building block for more complex programs
- If you want a bunch of blocks and only one to run, can you elif



```python
x = 12

# If/else
if (True):
    print("Is true") # spacing is important in python

if (x == 12):
    print("Is true")

if (x == 15):
    print("Is true") # doesn't print
else:
    print("Is false")

if (x == 15):
    print("15")
elif (x == 12):
    print("12")
else:
    print("Other")

print()
```

```
Is true
Is true
Is false
12
```

Problems

Example Problem:

- Need to convert from numbers to letter the same way binary does (65 = A, 66 = B, etc.)
- Have the user input that number and the program output the corresponding letter

Practice Problem 1:

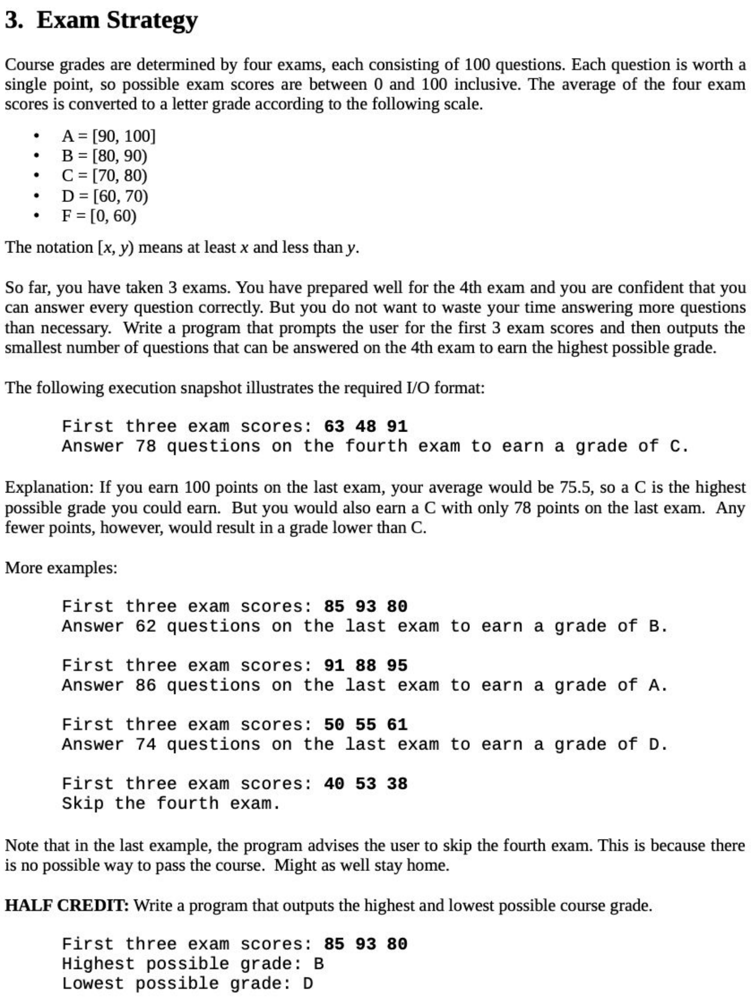
- Goal: Create a program that will find a launched object’s height after 10 seconds.
- Relevant Information:
  - Formula: x(t) = 0.5 * at^2 + vi*t + xi
  - A = -9.81 m/s^2
  - Vi = 0 m/s
  - T = 10 s
  - Xi = 0 m
  - Looking for x(10)

Practice Problem 2:

- Src: https://www.101computing.net/eureka-and-king-hieros-crown/
- Goal: Write a program that can determine the material of a crown.
- Requirements:
  - Ask the user to input the mass (kg) and volume (m^3) of a crown and calculate the density to determine the material used.
- Relevant Information:
  - Densities in kg/m^3 -
    - Aluminum = 2400-2700; Bronze = 8100-8300; Silver = 10,400-10,600; Lead = 11,200-11,400; Gold = 17,100-17,500; Platinum = 21,000-21,500
  - Tests -
    - Silver crown: 0.567 kg & 0.000054 m^3; Gold crown: 1.213 kg & 0.00007 m^3
    - Lead crown: 0.731 kg & 0.000065 m^3; Bronze crown: 0.585 kg & 0.000071 m^3

Practice Problem 3:

Can input on 3 lines instead of 1



3. Exam Strategy

Course grades are determined by four exams, each consisting of 100 questions. Each question is worth a single point, so possible exam scores are between 0 and 100 inclusive. The average of the four exam scores is converted to a letter grade according to the following scale.

- A = [90, 100]
- B = [80, 90)
- C = [70, 80)
- D = [60, 70)
- F = [0, 60)

The notation [*x*, *y*) means at least *x* and less than *y*.

So far, you have taken 3 exams. You have prepared well for the 4th exam and you are confident that you can answer every question correctly. But you do not want to waste your time answering more questions than necessary. Write a program that prompts the user for the first 3 exam scores and then outputs the smallest number of questions that can be answered on the 4th exam to earn the highest possible grade.

The following execution snapshot illustrates the required I/O format:

```
First three exam scores: 63 48 91
Answer 78 questions on the fourth exam to earn a grade of C.
```

Explanation: If you earn 100 points on the last exam, your average would be 75.5, so a C is the highest possible grade you could earn. But you would also earn a C with only 78 points on the last exam. Any fewer points, however, would result in a grade lower than C.

More examples:

```
First three exam scores: 85 93 80
Answer 62 questions on the last exam to earn a grade of B.

First three exam scores: 91 88 95
Answer 86 questions on the last exam to earn a grade of A.

First three exam scores: 50 55 61
Answer 74 questions on the last exam to earn a grade of D.

First three exam scores: 40 53 38
Skip the fourth exam.
```

Note that in the last example, the program advises the user to skip the fourth exam. This is because there is no possible way to pass the course. Might as well stay home.

HALF CREDIT: Write a program that outputs the highest and lowest possible course grade.

```
First three exam scores: 85 93 80
Highest possible grade: B
Lowest possible grade: D
```

The End:

- If anyone wants to continue working or get feedback, you can ask either of us
- There will be a GitHub (https://github.com/battler82004/jt-prgrm-club-22-23) where we'll post solutions and the notes from the lesson
  - Will hopefully uploads solutions on Thursdays
  - To contact James: James.Taddei@jtasd.org
  - To contact Jon: Jonathan.Duffy@jtasd.org
- Next week:
  - Hoping to look at loops and string manipulation next week