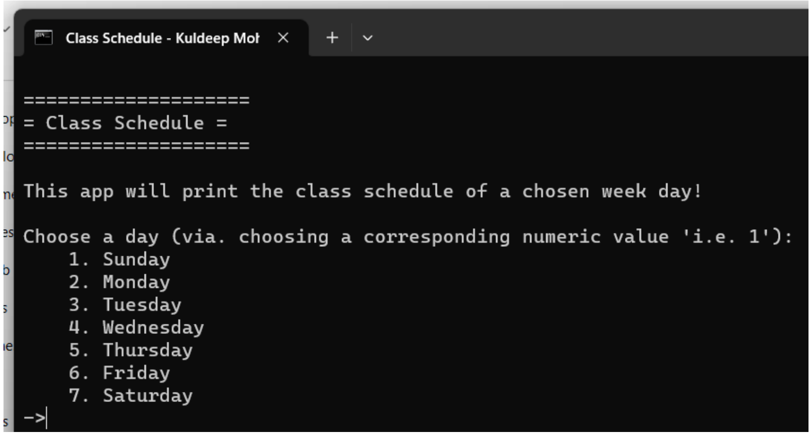
Kuldeep Mohanta
100656950
April 9, 2023

ICE 10 – Enums

Part 2 – Initial Screen

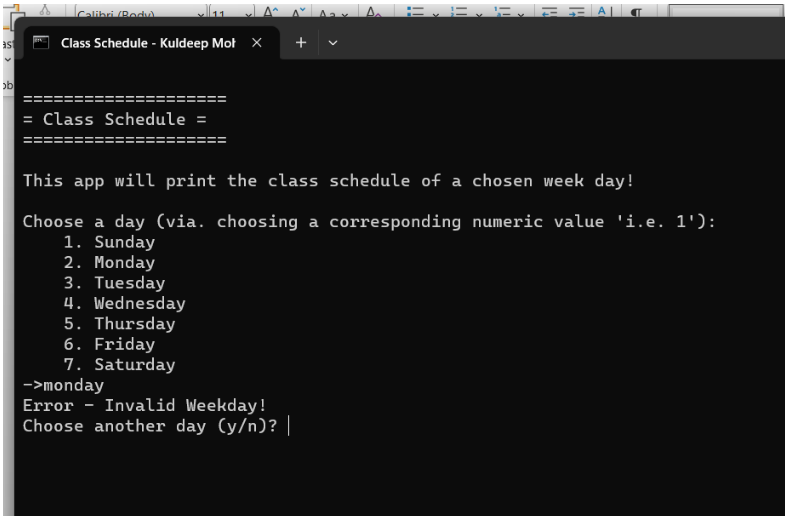


```
====================
= Class Schedule =
====================

This app will print the class schedule of a chosen week day!

Choose a day (via. choosing a corresponding numeric value 'i.e. 1'):
    1. Sunday
    2. Monday
    3. Tuesday
    4. Wednesday
    5. Thursday
    6. Friday
    7. Saturday
->
```

Screenshot 1

Context: As the application launches, the banner of the application is displayed with the program's introduction and a short description of what the program will present to the user. It prompts the user to input a day with instructions on what kind of input to provide.

Part 3 – Input Screen and Validation

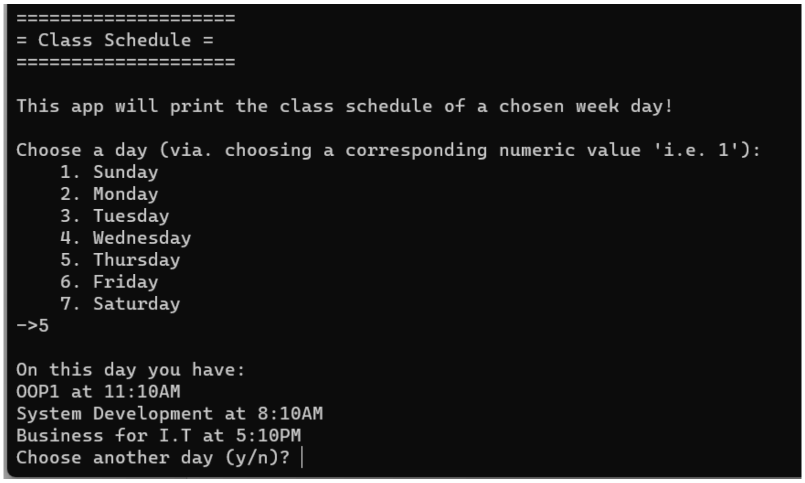


```
====================
= Class Schedule =
====================

This app will print the class schedule of a chosen week day!

Choose a day (via. choosing a corresponding numeric value 'i.e. 1'):
    1. Sunday
    2. Monday
    3. Tuesday
    4. Wednesday
    5. Thursday
    6. Friday
    7. Saturday
->monday
Error - Invalid Weekday!
Choose another day (y/n)?
```

Screenshot 2

Context: The user had entered an invalid input to choose the weekday (as they need to enter a numeric value to choose a weekday as instructed) so the program displays an error.

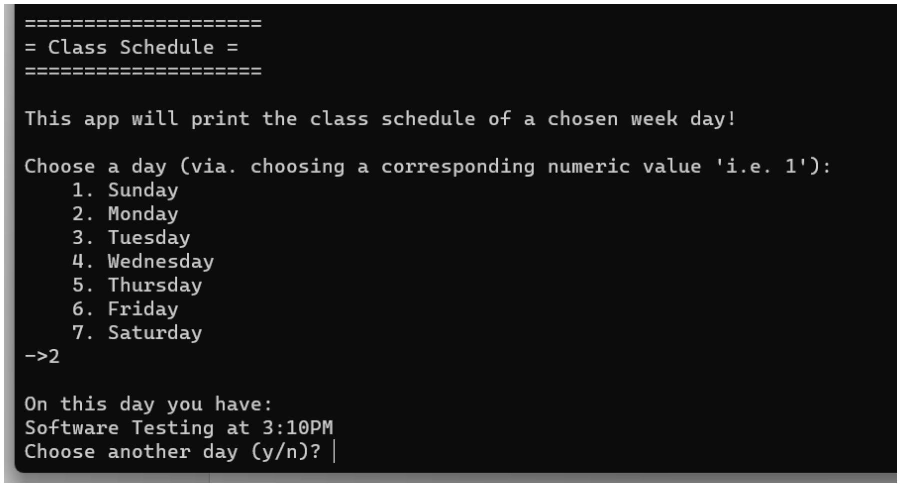
The app then prompts the user if they would like to input again.



```
====================
= Class Schedule =
====================

This app will print the class schedule of a chosen week day!

Choose a day (via. choosing a corresponding numeric value 'i.e. 1'):
    1. Sunday
    2. Monday
    3. Tuesday
    4. Wednesday
    5. Thursday
    6. Friday
    7. Saturday
->5

On this day you have:
OOP1 at 11:10AM
System Development at 8:10AM
Business for I.T at 5:10PM
Choose another day (y/n)?
```

Screenshot 3

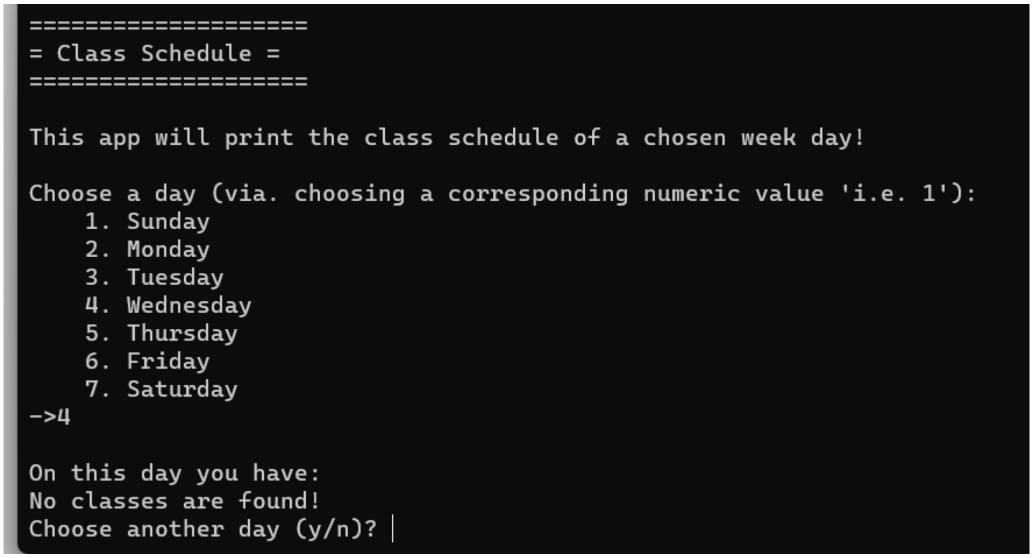
Context: The user had entered correct (valid) input to find a corresponding schedule for the day chosen, in this case the user has 3 classes in the system for option ("5") aka. Thursday. After which, the user is prompted if they would like to choose another day.



```
====================
= Class Schedule =
====================

This app will print the class schedule of a chosen week day!

Choose a day (via. choosing a corresponding numeric value 'i.e. 1'):
    1. Sunday
    2. Monday
    3. Tuesday
    4. Wednesday
    5. Thursday
    6. Friday
    7. Saturday
->2

On this day you have:
Software Testing at 3:10PM
Choose another day (y/n)?
```

Screenshot 4

Context: The user had entered correct (valid) input to find a corresponding schedule for the day chosen, in this case the user has 1 class in the system for option ("2") aka. Monday. After which, the user is prompted if they would like to choose another day.



```
====================
= Class Schedule =
====================

This app will print the class schedule of a chosen week day!

Choose a day (via. choosing a corresponding numeric value 'i.e. 1'):
    1. Sunday
    2. Monday
    3. Tuesday
    4. Wednesday
    5. Thursday
    6. Friday
    7. Saturday
->4

On this day you have:
No classes are found!
Choose another day (y/n)?
```

Screenshot 5

Context: The user had entered correct (valid) input to find a corresponding schedule for the day chosen, in this case the user has 0 classes in the system for option (“4”) aka. Wednesday. After which, the user is prompted if they would like to choose another day.

Part 5 – Answer the Questions

Question 1. What is an enum?

Enum is a data type within Java used to store a fixed set of constants. Its unique nature of storing constants can help preserve the readability and overall maintenance of a program since values are pre-determined and not hard-coded, which is (in many cases) always a positive gain!

Question 2. When should we use an enum instead of constants?

With constants, usually, they are determined for a singular purpose. While with enums, there can be a set of similar values. With enums, it allows constants intended to be used for similar purposes to be grouped together, which will increase overall code integrity and readability in the long run. It is beneficial to consider using enums for programs that utilize many constants.

Question 3. What is the difference between an enum and a class?

With enums, it is a unique type of class that defines a fixed set of constants, but unlike a class, it does not have anything to do with creating a blueprint from which objects can be created.

Enums also have specific restrictions such as:

- Their inability to extend (inheritance), they need to use only private constructors.
- While a class can have many methods or fields, an enum can have methods and fields, but those only apply to fixed instances (which come from the private constructors that create the fixed set of instances)