

LCS-ES Group Meeting #6

Exercise 1: Using methods

This exercise will help you to understand classes and the services they can provide.

Part (a)

Write a UML `Book` class to be used in a local books store. This is rather open ended, so feel encouraged to make reasonable assumptions.

Part (b)

Download the partial `Book` class and complete the code by completing the `sellBook` method. Once you are done, predict the output based on the main method provided, and then run the program.

Part (c)

Replace the main method with a Java translation of the following

- 1: prompt user for choice
- 2: if choice == 1 quit
- 3: if choice == 2 sell a copy
- 4: if choice == 3 stock a shipment
- 5: print current 'inventory'
- 6: goto 1

Part (d)

Assume that the following (nonsense) code is in class `BookMain` (not in the `Book` class) and find all of the errors. Suggest a plausible correction for each error.

```
Book oneBook = new Book("Harry Potter", "Rowlings", 19.95);
oneBook.sellCopy;
if (oneBook.overStocked())
{
    oneBook.lowerPrice();
}
```

Exercise 2: If and Switch Statements

Use the yellow and pink cards for this exercise. Each yellow card has a code fragment that includes an if statement, and each pink card has a code fragment that includes a switch statement. Working in pairs, match each if-fragment with one or more equivalent switch-fragments (some switch fragments have no matching if-fragment). Assume all variables are declared as integers except those that hold characters.

Be careful, some are tricky!

Exercise 3: Programming with Switch Statements

For this exercise, working in pairs, you will write a program that displays the favorite color, or the birthday, or the number of siblings of someone in your pair or one of their imaginary friends (have the program know about 4 people). Here's an example of what will happen when you run the program.

- First, the program will display:
Whose information would you like to access?
Enter A, B, C, or D to choose a person, or press Q to exit.
A) Bob
B) Jill
C) Mike
D) Ashley
Enter your choice now:
- After you enter A, B, C, or D, the program will display all the desired information; for example, if you chose 'C' for Mike, the program will print Mike's favorite color, his birthday and the number of Mike's siblings.

Part (a)

Start by defining a Person class to store one person's name, favorite color, birthday, and number of siblings (you might consider several different ways to represent the birthday). Favor the use of switch statements in your code.

Write a constructor that has four arguments (the four pieces of information about the person). Also write 4 accessor methods that return the name, favorite color, birthday, or number of siblings. Write 4 mutator methods that permit a change of the name, color, birthday or number of sibs.

Write a main method to test your Person class. Create one Person object for each person in your group, then print all of the information for each person.

Part (b)

Now change the main method so that after creating one Person for each person in your group, it prints the menu shown above (that allow the user to select one person). (Hint: To print the menu, use the getName method for each Person object). Make use of switch statements that switch on the letter typed in to decide which Person object to use, and which of its methods to call.

ON YELLOW PAPER

```
if (x == 0)
    y = 1;
else if (x == 1)
    y = 2;
else
    y = 3;
```

```
if (x == 0)
    y = 1;
else if (x == 1)
    y = 2;
```

```
if (x == 4)
{
    x = x + 4;
}
else if (x <= 7 && x < 4)
{
    x = x + 7;
}
```

```
if (x == 1 || x == 3)
{
    System.out.println
        ("Good Morning!");
}
else
{
    System.out.println
        ("Good Afternoon!");
}
```

```
if (character != ' ')
    Action1();
else
    Action4();
```

```
if (x == 8)
    x = 0;
else if (x*2 == 8)
    x = 2;
else if (x - 1 == 13)
    x = 3;
```

```
if (c > 2 && c < 4)
    c = 10;
else if (c <= 0 || c > 4)
    c = 100;
```

```
if (i > 0 && i < 4)
    x = 10;
else if (i <= 0 || i > 7)
    x = 40;
else if (i == 4)
    x = 20;
else if (i > 4 && i <= 7)
    x = 30;
```

ON PINK PAPER

```
switch (x)
{
    case 0: y = 1;
           break;
    case 1: y = 2;
           break;
    default: y = 3;
}
```

```
y = 3;
switch (x)
{
    case 0: y = 1;
           break;
    case 1: y = 2;
           break;
}
```

```
switch (x)
{
    case 0: y = 1;
           break;
    case 1: y = 2;
           break;
}
```

```
switch (x)
{
    case 0: y = 1;
           break;
    case 1: y = 2;
           break;
    default: y = 0;
}
```

```
switch (x)
{
    case 5:
    case 6:
    case 7: x = x + 7;
           break;
    default: x = x + 4;
}
```

```
switch (x)
{
    case 1:
    case 3: System.out.println
            ("Good Morning!");
           break;
    default: System.out.println
            ("Good Afternoon!");
}
```

```
switch (x)
{
    case 1:
    case 3: System.out.println
            ("Good Morning!");
}
System.out.println
("Good Afternoon!");
```

```
switch (character)
{
    case ' ': Action4(); break;
    default: Action1();
}
```

```
switch (character)
{
    case ' ': Action1(); break;
    default: Action4();
}
```

```
switch (x)
{
    case 8: x = 0; break;
    case 4: x = 2; break;
    case 14: x = 3; break;
}
```

```
switch (x)
{
    case 8: x = 0;
    case 4: x = 2;
    case 14: x = 3;
}
```

```
switch (c)
{
    case 1:
    case 2:
    case 4: break;
    default: c = 100;
}
```

```
switch (c)
{
    case 3: c = 10; break;
    default: c = 100;
}
```

```
switch (i)
{
    case 1:
    case 2:
    case 3: x = 10; break;
    case 4: x = 20; break;
    case 5:
    case 6:
    case 7: x = 30; break;
    default: x = 40;
}
```

```
switch (x)
{
    case 4: x = x + 4; break;
    case 5:
    case 6:
    case 7: x = x + 7; break;
}
```