

Java Objects Part 2 Notes

Team Treehouse

May 22, 2020

1 constants

- Are named *IN_CAPITALIZED_SNAKE_CASE*
- Can be done using *static* keyword
- Allows variables and methods to be expoused without instantiation

```
1 public class PezDispenser {  
2     public static final int MAX_PEZ = 12; // <- 1. static declared  
3     here :)  
4     ...  
5 }
```

Listing 1: lesson_1/PezDispenser.java

```
1 import java.io.Console;  
2  
3 public class Example {  
4     public static void main(String[] args) {  
5         ...  
6         System.out.printf("FUN FACT: There are %d PEZ allowed in  
7         every dispenser\n", PezDispenser.MAX_PEZ); // 2. <- And is used  
8         here :)  
9         ...  
10    }  
11 }
```

Listing 2: lesson_1/Example.java

Notes:

- Files can be compiled and displayed by typing *javac Example.java* && *java Example* in terminal

2 Filling the Dispenser

- *void* keyword means nothing is returned at the end of a method

```
1     public class PezDispenser {
2         public void fill() { // <- This little guy here :)
3             this.pezCount = MAX_PEZ;
4             System.out.printf("The current count of delicious PEZ is %
5                 d\n", this.pezCount);
6         }
7     }
```

Listing 3: lesson_3/PezDispenser.java

```
1     import java.io.Console;
2
3     public class Example {
4         public static void main(String[] args) {
5             ...
6             dispenser.fill(); // <- 2. Is used like this
7         }
8     }
9
10
```

Listing 4: lesson_3/Example.java

Notes:

- Files can be compiled and displayed by typing *javac Example.java* && *java Example* in terminal
- Always start with private methods, and turn to public when needed.