

CSE 142 Final Cheat Sheet

Syntax templates:

Declaring and using arrays

```
type[] name = new type[length];  
type[] name = {VAL1, VAL2, VAL3, ...};  
name[index] = value;  
name.length // number of elements in array
```

Declaring objects

```
Type objectName = new Type(parameters);
```

Classes

Field (data inside each object)

```
private type name;
```

Method (behavior inside each object)

```
public type name(parameters) {  
    statement(s);  
}
```

Constructor (code to initialize new objects)

```
public className(parameters) {  
    statement(s);  
}
```

toString method (called when an object is printed)

```
public String toString() {  
    code that produces/returns a String;  
}
```

Inheritance

```
public class name extends superclass {  
    field(s), constructor(s), method(s), etc.  
}
```

Critter class template:

```
public class name extends Critter {  
  
    field(s)  
  
    constructor(s)  
  
    public boolean eat() {  
        statement(s) that return either true (eat) or false (don't eat)  
    }  
  
    public Attack fight(String opponent) {  
        statements(s) that return either Attack.ROAR, Attack.POUNCE, or Attack.SCRATCH  
    }  
  
    public Color getColor() {  
        statement(s) that return a Color  
    }  
  
    public Direction getMove() {  
        statement(s) that return either Direction.NORTH, Direction.SOUTH, Direction.EAST, Direction.WEST, or Direction.CENTER  
    }  
  
    public String toString() {  
        statement(s) that return a String  
    }  
}
```

| Random Method | Description |
|---------------------------|----------------------------------|
| <code>nextInt(max)</code> | random integer from 0 to $max-1$ |

| Math Method | Description |
|--------------------------------|----------------------------|
| <code>Math.abs(value)</code> | absolute value |
| <code>Math.min(v1, v2)</code> | smaller of two values |
| <code>Math.max(v1, v2)</code> | larger of two values |
| <code>Math.round(value)</code> | nearest whole number |
| <code>Math.sqrt(value)</code> | square root |
| <code>Math.pow(b, e)</code> | base to the exponent power |

| String Method | Description |
|---------------------------------------------|-----------------------------------------------------------------------------------|
| <code>contains(str)</code> | true if this string contains the other's characters inside it |
| <code>endsWith(str), startsWith(str)</code> | true if this string starts/ends with the other's characters |
| <code>equals(str)</code> | true if this string is the same as <i>str</i> |
| <code>equalsIgnoreCase(str)</code> | true if this string is the same as <i>str</i> , ignoring capitalization |
| <code>indexOf(str)</code> | index in this string where given string begins (-1 if not found) |
| <code>length()</code> | number of characters in this string |
| <code>replace(str1, str2)</code> | replace all occurrences in this string of <i>str1</i> with <i>str2</i> |
| <code>substring(i, j)</code> | characters in this string from index <i>i</i> (inclusive) to <i>j</i> (exclusive) |
| <code>substring(i)</code> | characters in this string from index <i>i</i> to end (inclusive) |
| <code>toLowerCase(), toUpperCase()</code> | a new string with all lowercase or uppercase letters |
| <code>charAt(i)</code> | returns char at index <i>i</i> |

| Scanner Method | Description |
|--------------------------------------------|-------------------------------------------------------------------------------|
| <code>nextInt(), hasNextInt()</code> | read/return input token as <code>int</code> ; test if reading will succeed |
| <code>next(), hasNext()</code> | read/return input token as <code>String</code> ; test if reading will succeed |
| <code>nextDouble(), hasNextDouble()</code> | read/return input token as <code>double</code> ; test if reading will succeed |
| <code>nextLine(), hasNextLine()</code> | read/return line as <code>String</code> ; test if reading will succeed |