IntelliJ Quick Reference

Useful Abbreviations

```
fori + tab          for (int i = 0; i < ; i++) {
                    }
psvm + tab          public static void main(String[] args){
                    }
sout + tab          System.out.println();
```

Useful Windows Shortcuts

```
Ctrl + Space        basic code completion
Shift + F6          rename
Ctrl + Alt + L      reformat code
Ctrl + /            comment / uncomment with line comment
Ctrl + Q            show documentation (JavaDoc)
```

Java Quick Reference

Console Input

```
Scanner input = new Scanner(System.in);
int intValue = input.nextInt();
long longValue = input.nextLong();
double doubleValue = input.nextDouble();
float floatValue = input.nextFloat();
String string = input.next();
String line = input.nextLine();
```

Console Output

```
System.out.println(anyValue);
System.out.print(anyValue);
System.out.printf("%4d %4.2f %4s", intValue, doubleValue, string);
```

Primitive Data Types

```
byte
short
int
long
float
double
char
boolean
```

Arithmetic Operators
```
+     addition
–     subtraction
*     multiplication
/     division
%     remainder
++var preincrement
--var predecrement
var++ postincrement
var-- postdecrement
```

Assignment Operators

```
=      assignment
+=     addition assignment
-=     subtraction assignment
*=     multiplication assignment
/=     division assignment
%=     remainder assignment
```

Relational Operators

```
<      less than
<=     less than or equal to
>      less than
>=     less than or equal to
==     equal to
!=     not equal
```

Logical Operators

```
&&     AND
||     OR
!      NOT
^      exclusive OR
```

If Statements

```
if (condition) {
    statements;
}

if (condition) {
    statements;
} else {
    statements;
}

if (condition1) {
    statements;
} else if(condition2) {
    statements;
} else {
    statements;
}
```

Switch Statement

```
switch (switchExpression) {
    case value1:
        statements;
        break;
    …
    case valueN:
        statements;
        break;
    default:
        statements;
}
```

Conditional Operator

```
booleanExpression? expression1 : expression2;
Example:
max = (number1 > number2) number1 : number2;
```

Loop Statements

```
while (condition) {
    statements;
}

do {
    statements;
} while (condition);

for (initiation; condition; adjustment) {
    statements;
}
```

Frequently Used Static Constants/Methods

```
Math.PI
Math.random()
Math.pow(a, b)
Math.abs(a)
Math.min(a, b)
Math.sqrt(a)
Arrays.sort(type[] list)
Arrays.binarySearch(type[] list, type key)
```

Single Dimensional Arrays

```
int[] list = new int[10];
list.length;
int[] list = {1, 2, 3, 4};
```

Multi Dimensional Arrays

```
int[][] list = new int[10][10];
list.length;
list[0].length;
int[][] list = {{1, 2}, {3, 4}};
```

String Class

```
String string = "Welcome";
int length = string.length();
char character = string.charAt(index);
int compare = string.compareTo(anotherString);
boolean bool = string.equals(anotherString);
boolean bool = string.startsWith(anotherString);
boolean bool = string.contains(anotherString);
String string2 = string.trim();
String string2 = string.toUpperCase();
String string2 = string.toLowerCase();
int index = string.indexOf(character);
int index = string.lastIndexOf(character);
String string2 = string.substring(beginIndex);
String string2 = string.substring(beginIndex, endIndex);
```

ArrayList Class

```java
ArrayList<E> list = new ArrayList<>();
list.add(object);
list.add(index, object);
list.clear();
Object object = list.get(index);
boolean bool = list.isEmpty();
boolean bool = list.contains(object);
int size = list.size();
list.remove(index);
list.set(index, object);
int index = list.indexOf(object);
int index = list.lastIndexOf(object);
```

LocalDate Class

```java
LocalDate date = LocalDate.now();
LocalDate date = LocalDate.of(year, month, dayOfMonth);
LocalDate date2 = date.plusYears(yearsToAdd);
LocalDate date2 = date.plusMonths(monthsToAdd);
LocalDate date2 = date.plusWeeks(weeksToAdd);
LocalDate date2 = date.plusDays(daysToAdd);
String string = date.toString();
int compare = date.compareTo(anotherDate);
boolean bool = date.equals(object);
boolean bool = date.isBefore(anotherDate);
boolean bool = date.isAfter(anotherDate);
int day = date.getDayOfMonth();
int month = date.getMonthValue();
Month month = date.getMonth();
int year = date.getYear();
DayOfWeek dayOfWeek = date.getDayOfWeek();
```