Primitive types (2.1)

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>int</td>
<td>integers</td>
<td>42, -3, 92851</td>
</tr>
<tr>
<td>double</td>
<td>real numbers</td>
<td>3.14, 2.0</td>
</tr>
</tbody>
</table>
| char    | a character of text | 'a', 'X', '
'         |
| boolean | logical values      | true, false           |

Expressions (2.1)

- precedence: () before */% before +-
- with int, / is integer quotient and % is integer remainder
- Strings can be concatenated with other values

```
1 * 2 + 3 * 5 / 4   # $5 + 9.0 / 4.0 + 1
2 + 3 * 5 / 4       # $2.25 + 1
2 + 15 / 4          # $2.25 + 1
2 + 3               # "$2.251"
```

Variables (2.2)

```
type name;  // declaration (creates a variable but doesn't give it any value)
name = value; // assignment (stores a value into a variable)
type name = value; // declaration/initialization (creates a variable and stores a value into it)
```

```java
int x;
int y = 3;
x = 1 + y * 2;    // x stores the value 7
```

The for loop (2.3)

```
for (initialization; test; update) {
    statement;
    statement;
    ...
    statement;
}
```

```java
for (int i = 1; i <= 10; i++) {
    System.out.println(i + " squared is " + (i * i));
}
```

Nested for loops (2.3)

```
for (int line = 1; line <= 5; line++) {
    for (int j = 1; j <= (-1 * line + 5); j++) {
        System.out.print(".");
    }
    System.out.println(line);
}
```

Class constants (2.4)

```
public static final type name = value;
public static final int DAYS_PER_WEEK = 7;
```