

Universidade de Brasília – Faculdade UnB Gama
Disciplina: Técnicas de Programação
Professor: Maurício Serrano
Período: 2º/2015
Data de entrega: 24 de Agosto de 2015
Estudantes:

11/0014863 - Kleber Brito Moreira
11/0017561 - Mateus Furquim
11/0017692 - Matheus Mello Nascimento
11/0064879 - Nicolas Boarin

Stylesheet - Grandes Pontos

1. Comments

Comments must obey the following pattern:

1.a) One-line comments must be written using a double slash ‘//’ before the comment, followed by a space ‘ ’ and a capital letter. Do not finish it with a period;

```
// This is the comment pattern
```

1.b) Comments in blocks must be written between the ‘/*’ and ‘*/’. For each new line, there must have an asterisk aligned with the previous one;

```
/**  
 * This is  
 * the pattern  
 * to be followed  
 */
```

1.c) Comments which precede methods must be written in blocks explaining its use and its return:

```
/**  
 * The method calculatePerimeter calculates the perimeter of a triangle  
 * using its sides.  
 * @return the triangle's perimeter  
 */  
function calculatePerimeter( side1, side2, side3 ) {  
    var result = side1+side2+side3;  
    return result;  
}
```

1.d) Comments explaining one line of the code must precede the line; boolean equality;

```
// Is the side 1 equal to side 2?  
var equality = (side1 == side2);
```

1.e) Comments exceeding 80 characters must be broken into multiple lines following the block pattern;

```
/* This is a coment which exceeds 80 characters. This is a coment which exceeds
 * 80 characters. This is a coment which exceeds 80 characters. This is a
 * coment which exceeds 80 characters.
 */
```

2. Types and Names

2.a) Names of Classes, Attributes, and Methods must follow the CamelCase pattern, which consists in capitalizing the initial letters of each word.

2.b) Classes names initiates with capital case.

```
var MyClass = new MyClass( name );
```

2.c) Attributes names initiates with lower case.

```
var personAge;
```

2.d) Methods names initiates with lower case.

```
function setPersonName( name );
```

2.e) Constants names must be capitalized and use underscore between names.

```
const MAX_PERSON_AGE = 120;
```

2.f) To reduce the scope, a variable which is used only in the current scope must be declared as let:

```
let auxiliar = "This variable will vanish once it reaches the end of the scope";
```

2.g) Only one variable must be declared per line.

3. Strings

3.a) Strings must be written between double quotes “”;

```
var string = "This is a string example written between double quotes";
```

4. Indentation

4.a) The indentation must be written by two (2) spaces or an equivalent tab.

```
function getMethod() {  
  --// ...  
  --if ( condition1 == condition2 ) {  
    ----// ...  
  --}  
}
```

5. Braces

5.a) Opening braces must be used on the same line of the block structure, after an empty space;

5.b) Closing braces must be written aligned with the statement which opened the block;

```
var myObject = MyClass( parameter1, parameter2 ) {  
  | function getMethod() {  
  | | // ...  
  | | if ( condition1 == condition2 ) {  
  | | | // ...  
  | | }  
  | }  
  | while ( condition3 == condition4 ) {  
  | | // ...  
  | | for ( var i = 0; i < count; i++) {  
  | | | // ...  
  | | }  
  | }  
}
```

6. Classes

Classes must be according to the following model:

6.a) There must have a blank line after the class signature.

```
function StyleSheet() {  
  
  /**  
   * There must have a blank line between  
   * the class signature and the  
   * first line of code of the class  
   */  
}
```

6.b) Prototypes must be used as inheritance:

```
function Person( name ) {
    var name = name;
}

PhysicalPerson.prototype = new Person();

var physicalPerson = new PhysicalPerson( "Nome da Pessoa" );
console.log( physicalPerson.name ); // Nome da Pessoa
```

7. Control Structure: if

- 7.a) There must have a blank space immediately after the keyword if.
- 7.b) There must have a blank space between the parentheses and the if condition.
- 7.c) The comparison operators must have a blank space immediately before and after.
- 7.d) If the line exceeds 80 characters, it must be broken into multiple lines and the condition must be aligned with the previous one.
- 7.e) The body must be written between braces, even when it is a single line.
- 7.f) Blocks of Else if or else must be initiated on the same line of the closing braces.
- 7.g) There must have an else block after all if or else if block.

```
if ( this.personAge < 18 && /* There is more than 80 characters here */
    this.personVIP == NULL) {
    alert( "Not authorized to drive!" );
} else {
    // . . .
}

if ( weekDay == 1 ) {
    console.log( "Sunday" );
} else if ( weekDay == 2 ) {
    console.log( "Monday" );
} else {
    // ...
}
```

8. Control structure: while

- 8.a) While follows the if pattern:

```
while ( count < numElements ) {
    sumElements = sumElements + 1;
}
```

9. Control structure: for

9.a) The `for` structure follows the `if` pattern.

9.b) Semicolon must be placed immediately after the assignment and the comparison, and an empty space after it.

```
for ( int i = 0; i < numElements; i++ ) {  
    console.log( numElements );  
}
```

10. Control structure: switch

a. Defaults and cases must not be indented;

10.b) Case blocks and default blocks must be indented.

10.c) `break` must be indented as well.

```
switch ( dayWeek ) {  
case 1:  
    console.log( "Sunday" );  
    break;  
  
case 2:  
    console.log( "Monday" );  
    break;  
  
// ...  
  
default:  
    console.log( "" );  
    break;  
}
```

11. Treatment of Exceptions

11.a) The `try`, `catch`, and `finally` blocks should be on their own line, after each closing braces.

```
try {  
    / block to be executed  
}  
catch( exception ) {  
    // chatch's instructions block  
}  
finally {  
    // finally instructions block  
}
```

12. Coding language

The coding language for programming techniques applying is English.