Types of Errors

Syntax Errors

"Syntax" is a term used to describe the rules of a language, and a syntax error is where we don't follow the rules of the language, so for example, in English (this is a speaking language or a "natural language") the phrase "*the cat sat on the mat*" follows the rules of the language, because in English we can have a noun ("cat") followed by a verb ("sat") followed by a preposition ("on"), followed by a noun ("mat"). So the chain *noun-verb-preposition-noun* follows the rules. However the phrase "*sat the cat the mat on*" is not legal because you the chain *verb-noun-noun-preposition* does not follow the rules of English grammar. In a programming language it's the same thing, the phrase "X = 5" is legal, but "**X 5 =**" isn't legal, and the computer won't understand what it means. Syntax errors are easy to locate because the computer will give an error when you run the program.

Semantics Errors

A semantic error is one where the syntax is correct, but it breaks some other rule in the programming language, and will not compile, so, for example, a type error:

```
x = "Hello, World!"
print(x+1)
```

Or using a variable before declaring it:

```
print(x)
x = "Hello, World!"
```

There are a number of typical semantic errors, and they usually revolve around a program that uses some resource before telling the program we want to use it.

Logical Errors

Logical errors don't give us any errors when we compile our programs, but they do give us the wrong answer, so if we wrote a program to calculate the area of a circle:

```
AreaOfACircle = 3.1416 * Radius
```

This is wrong, because the area of a circle is Pi * R^2, which should be written as:

```
AreaOfACircle = 3.1416 * Radius * Radius
```

So the first program will give us the wrong answer, but won't produce a compilation error. A common logical error on conditions of IF statements, where we mix up the "less than" ("<") and "greater than" (">"), so for example:

```
x = int(input("Input a value: "))
if (x > 5):
    print("X is less than 5")
# EndIf;
```

This program won't give a compilation error, but it will give the wrong answer.