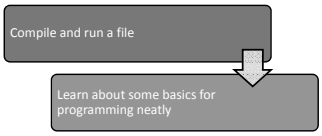
Writing your first program

Objectives



Your first program

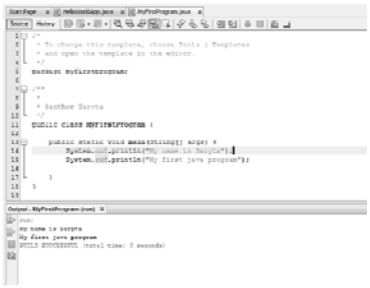
- Open Netbeans
- Create a new project
  - File → New Project
    - Make sure *Java* is selected for Categories and *Java Application* is selected for Projects
  - Select Next
  - Enter *MyFirstProgram* under Project Name
    - Make sure Create Main Class is selected
  - Select Finish
    - By default, MyFirstProgram.java should be highlighted and visible in the source window.

- Enter in the main method the line:
  - System.out.println("My first java program");
  - Make sure you enter it below the TODO line
- If you see a red line then there is an error.
- Now compile and run your code
  - Either hit the Run Project button
  - Hit F6
  - Under the run menu select *Run Project*

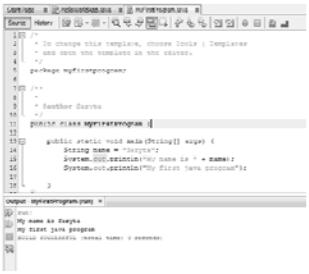
Note …

- There are a lot of things going on in the program we just wrote that you won't understand yet. The most important thing you need to know is that when Java sees a file like this, it always checks if there is a method called *main*. If there is, it uses it as the "default" method to run, and executes the code inside. That is why the statement *System.out.println("My First Java Program");* is executed when you run this code.
- For the most part, we will be using *main*()

- Let's try changing the program. Let's add a line that prints your name to the file. Be careful not to change the first few lines of the file, but you can insert your code as a new line after the beginning of main.
- Did that work? If not, you probably didn't notice a very small detail -- almost every line of ends with a semicolon (;). Your line needs to as well or it won't compile. If you got that right, good for you for noticing! This is one of those funny things that computers really care about, another syntax error.
- Here's how your file should have looked with the new line in it:
  - I deleted the TODO comment



- Let us modify our first program to use variables.
- Instead of entering our name, let us store it in a variable.
- Create a new variable of type String that will hold your name
  - String name = “Saryta”;
- Now, to display the variable with our name in it, we need to concatenate it to the output.
  - System.out.println(“My name is ” + name);

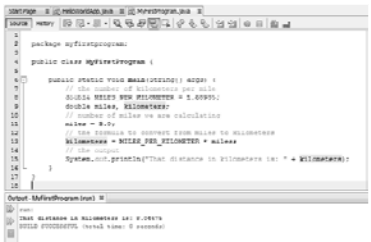


Simple program

- Let us add some more code to our first program.
- Our objective will be to convert miles into kilometers using variables.
- What variables do you think we might need to do this?
  - miles: how many miles we want to convert to kilometers.
  - kilometers: the miles converted into kilometers.
- Another useful piece of information might be how many kilometers there are in a mile.
- Finally, we need to do a calculation and display the value of kilometers.
- Can you think of a way to do this?

- You can do this right in your current project, or create a new one.
- Here is a solution:

```
public static void main(String[] args) {
    double MILES_PER_KILOMETER = 1.60935;
    double miles, kilometers;
    miles = 5.0;
    kilometers = MILES_PER_KILOMETER * miles;
    System.out.println("That distance in kilometers is: " + kilometers);
  }
```

Writing Legible Code

- Code
  - (definition) Code is the name for the words that you write when you program. Another word for code is "a program". Another word for programming is "coding." Code is written in a specific language (a "programming language"). A program, or a piece of code, gives instructions to the computer that it can understand and execute.
- Comment
  - (definition) A comment is some text in your program that explains how the program works to humans but is ignored by the computer. You can write a comment in two ways. Either put // right at the start of each line of a comment, or use /* to start your comment and */ to end your comment, with anything you want over multiple lines in between.

- From here on out, you should write code not only to be functional, but also to be legible. Programming languages are designed to bridge the gap between people and computers in both directions.
- However, neither computers, nor people, will understand clearly what your code is meant to do unless you work to make your code legible.

Some Rules for writing legible code

- Always comment your code
  - Comments are for humans only, they are ignored completely by the computer. Some people write a program out in comments, with explanations, before they even write a line of code! In your code, each piece of logic should be explained in comments.
- Always use meaningful variable names
  - If you had three kids, you would not name them "a" "b" and "c". Worse, imagine if your spouse was named "d". No one would know who you were talking about or what kind of thing they were (an adult? a child? a person?). Most programs that get written eventually is read by someone else! Your programs need to be legible to people (including yourself after you've forgotten about it), not just computers.

Some Rules for writing legible code

- Variable names start in lowercase, objects in uppercase
  - Variable names should start in lowercase ("bill", not "Bill"). If you use multiple words, you may capitalize each subsequent word ("billAndTip"). This helps to differentiate variables from other names in your code, such as objects, which start in uppercase.
- Indent your code properly
  - Indenting your code properly greatly increases legibility. A good rule of thumb for indenting code is to add a level of indentation within a set of curly braces ({}).

Summary

- We wrote some example programs and executed them. We used the main() method to cause something to be printed to the screen.
- We also learned about writing legible code, including the importance of commenting and naming