# Homework: C Data Types

This document defines the homework assignments from [the "C Programming" Course @ Software University](https://softuni.bg/trainings/1212/C-Programming-October-2015). Please submit as homework a single zip / rar / 7z archive holding the solutions (source code) of all below described problems.

## Declare Variables

Declare five variables choosing for each of them the most appropriate of the types char, short, unsigned short, int, unsigned int, long, unsigned long to represent the following values: 52130, **8942492113**, -115, 4825932, 97, -10000. Choose a large enough type for each number to ensure it will fit in it.

## Float or Double?

Which of the following values can be assigned to a variable of type float and which to a variable of type double: 34.567839023, 12.345, 8923.1234857, 3456.091? Write a program to assign the numbers in variables and **print** them to ensure no precision is lost.

## Variable in Hexadecimal Format

Declare an integer variable and assign it with the value 254 in hexadecimal format (0x##). Use a calculator online to find its hexadecimal representation. Print the variable and ensure that the result is "254".

## Gender

Declare a variable called isFemale and assign an appropriate value corresponding to **your gender**. Print it on the console.

|  |
| --- |
| **Expected Output** |
| true |

## Names

Declare two **string (char array) variables** holding your first name and last name. Print them in the console (mind adding an interval between them).

|  |
| --- |
| **Sample Output** |
| Mark Twain |

**Tip**: Make sure you properly declare the strings by specifying a correct size. Avoid buffer overflows.

## Quotes in Strings

Declare a string variable and assign to it the following value:

|  |
| --- |
| The "use" of quotations causes difficulties. \n, \t and \ are also special characters. |

Do the above in two different ways: with and without using **quoted strings**. Print the variables to ensure that their value was correctly defined.

|  |
| --- |
| **Expected Output** |
| The "use" of quotations causes difficulties. \n, \t and \ are also special characters. |

## Exchange Variable Values

Declare two integer variables a and b and assign them with 5 and 10 and after that exchange their values by using some programming logic. Print the variable values before and after the exchange.

|  |
| --- |
| **Expected Output** |
| Before:  a = 5  b = 10  After:  a = 10  b = 5 |

## Employee Data

A marketing company wants to keep record of its employees. Each record would have the following characteristics:

* First name
* Last name
* Age (0...100)
* Gender (m or f)
* Personal ID number (e.g. 8306112507)
* Unique employee number (27560000…27569999)

Declare the variables needed to keep the information for a single employee using appropriate primitive data types. Use descriptive names. **Print** the data at the console.

|  |
| --- |
| **Expected Output** |
| First name: Amanda  Last name: Jonson  Age: 27  Gender: f  Personal ID: 8306112507  Unique Employee number: 27563571 |

## Bank Account Data

A bank account has a **holder name** (first name, middle name and last name), available **amount of money** (balance), **bank name**, **IBAN** and **3 credit card numbers** associated with the account. Declare the variables needed to keep the information for a single bank account using the appropriate data types and descriptive names.

## \* Comparing Floats

Write a program that **safely compares floating-point numbers** (double) with precision eps = 0.000001. Note that we cannot directly compare two floating-point numbers a and b by a==b because of the nature of the floating-point arithmetic. Therefore, we assume two numbers are equal if they are more closely to each other than a fixed constant eps. Examples:

|  |  |  |  |
| --- | --- | --- | --- |
| **Number a** | **Number b** | **Equal (with precision eps=0.000001)** | **Explanation** |
| 5.3 | 6.01 | false | The difference of 0.71 is too big (> eps) |
| 5.00000001 | 5.00000003 | true | The difference 0.00000002 < eps |
| 5.00000005 | 5.00000001 | true | The difference 0.00000004 < eps |
| -0.0000007 | 0.00000007 | true | The difference 0.00000077 < eps |
| -4.999999 | -4.999998 | false | Border case. The difference 0.000001 == eps. We consider the numbers are different. |
| 4.999999 | 4.999998 | false | Border case. The difference 0.000001 == eps. We consider the numbers are different. |

## \* Print the ASCII Table

Find online more information about [**ASCII**](http://www.ascii-code.com/) (American Standard Code for Information Interchange) and write a program to prints the entire ASCII table of characters at the console (characters from 0 to 255). Note that some characters have a special purpose and will not be displayed as expected. You may skip them or display them differently. You may need to use for-loops (learn in Internet how).