Identifiers

Identifiers are the name which we give to variables, constants, functions or any sort of user-defined data (basically, any word which do not have a pre-defined meaning to the compiler).

These are the words which we use to name the variables we define to make it clear for our understandings. Once you move on to next program you can redefine the same word for something else. So, the meaning of that word holds for that piece of code only unless you import it.

In the example code provided in keyword section above, ***a*** was the identifier, name given to a variable which stores any value provided by the user.

There are some rules for naming ***identifiers: -***

- The first letter of your identifier must be a letter from English alphabet or an underscore (_) symbol.
- The other letters of your identifier can compose of English letters, numbers and underscore symbol.
- Identifier names are case-sensitive. So*, StaR* and *star* are treated as different.
- Some valid identifier names are: - your_name, _iam1, ty1rt, Ene, eNe
- Some invalid names are: - 1iam, der%er, rt@dd

So, this is all about identifiers. Hope you got a clear idea about identifiers now.

Important Points:

- Keywords are pre-defined functions while identifiers are names which we define.
- Keywords have a special meaning to the compiler which does not change from program to program but meaning of same identifier can change depending upon its usage in different programs.
- Keywords names are fixed but identifier’s name depends on our choice. There are however some rules to follow for naming.