**Salesforce Data Types**

Data types are used by Apex in Salesforce to describe the type of variable that the user is declaring. These data types are categorized into three types, they are:

* **Primitive –**

These data types are pre-defined type in the Apex programming language.

Examples include:

* **Id** – any 18-character value of Lightning Platform that is valid.

Example: Id id = ‘0038880452PALQ6’;

Here id is a 15-character record, it will be converted into 18-character by Apex.

All those values greater than 18 are treated invalid and left during runtime.

* **Integer** – a 32-bit value with no decimals.

Example: Integer i = 15;

The lowest value that an Integer can take is -2,147,483,648 and the highest value is 2,147,483,647.

* **Decimal** – a number with a decimal point.

Example: Decimal d = 15.36

* **Long** – a 64-bit value with no decimals.
* Example: Long l = 3625894310L

The lowest value that a Long can take is -263 and the highest value is 263. In the above example, L at the last of the value denotes that l is a long-valued number.

* **String** – any characters set that which is surrounded with single quotations.

Example: String s = ‘The women in the black forest and with a small lamp.’;

Though limitless number of characters can be used in Strings, Apex uses size limit of heap to make sure that programs don’t get too large.

* **Date** – a value that represents a day.

Example: Date d = Date.today();

Here d represents that day when the code will be executed.

The format for date is: YYYY-MM-DD

* Time – a variable that denotes the particular time.
* The format for time is: hh:mm:ss.sssZ
* **DataTime** – a value that denotes the day and time.

Example: DateTime dt = DateTime.now();

Here dt represents the date of the day when the code will be executed and the time.

The format for datetime is: YYYY-MM-DD hh:mm:ss (or) YYYY-MM-DDThh:mm:ssZ (or) YYYY-MM-DDThh:mm:ss.sssZ

* **Blob** - a single object that consists of binary data.
* **Boolean** – a variables which can take any of the three values ‘true’, ‘false’, and ‘null’.
* **sObject –**

The variables that are a collection of primitives, which represent a row of data are denoted as sObjects. These variables/ objects can only be declared using the Simple Access Protocol in Apex.

**Example:** Contact c = new Contact();

Here, c is an object that represents a collection of data.

* **Enum –**

It is an abstract data type for variables which take only a single set of values. The values specified do not have a particular numerical order in general.

The syntax of Constructor cannot be used for Enum types.

**Example:** public enum Deckofcards {Diamonds, Hearts, Clubs, Spades}

Here, the new data type Deckofcards is created using enum which has Diamonds, Hearts, Clubs, Spades as the variables.

**Field Types:**

Every field created in Salesforce Apex has a data type. Whenever a field is created, the developer is prompted to choose a data type for it. The available field data types are:

Address, Anyype, Calculated, Combobox, Currency, DataCategoryGroupReference, Email, EncryptedString, ID, JunctionIdList, Location, Masterrecord, Multipicklist, Phone, Percent, Picklist, Textarea, Reference, URL.

|  |  |  |
| --- | --- | --- |
| **Data Types** | **Field Types** | **Transformation Data Types** |
| ID | Look-up relational, Master-detail relationship | String |
| String | Auto-number, email, phone, multi-select picklist, text, text area, long text area, rich text area, data category group reference, and URL | String |
| Boolean | Checkbox | Integer |
| Double | Currency, formula, number, percent, and roll-up summary | Decimal |
| Varies by type | Currency, date, datetime, number, percent, text | String |

**Table-1: Field Types grouped with Data Types [1]**

From Table-1, we can understand that the available field types are grouped into the primitive data types based on their representation.

For example,

The email can be in the form of [abc@gmail.com](mailto:abc@gmail.com) (or) [vkm@yahoo.com](mailto:vkm@yahoo.com), as it is a collection of characters, it is transformed into String data type.

The checkbox is a value of true, false or null, it will be represented in the form of 0/1. So, it is transformed into Integer data type.

The value of percent can be as 63.5 (or) 100.0, as it is the representation of decimal value it is grouped into Decimal data type.

The phone numbers can sometimes include alphabets, so they can be String.

For a multi-select picklist, the values are String, for instance, they can be brand names of cars or ice cream flavors, etc.

Text areas are those places to write opinions of customers, or if they wish to express any feelings. This is usually in the form of alphabets and words. So, it is grouped as String.

**Conclusion:**

In conclusion, we can understand that Salesforce Apex, like all other programming languages, contain many data types that represent the type of the variables and the values to be stored in those variables.

It also includes many field types, which can be used when the time of declaration of fields like address, phone, text area, checkbox, etc. These system-defined field types make the work of developers easier and the users can choose any type based on their idea of declaring the variables.

With these flexibilities of having different data types and field types, it can be expected to have Salesforce Apex as the most widely being used programming platform to build projects by individuals and organizations in the coming future.

**References:**

1. <https://intellipaat.com/blog/tutorial/salesforce-tutorial/different-types-data-types-field-types/#1> – Blog on Salesforce Tutorial for Developers by Intellipaat

**Please leave the query and comments in the comment section.**