float Data Type

It is a 32-bit, single-precision IEEE 754 (Standard for Floating-Point Arithmetic) floating-point number. It means that it gives 6-7 decimal digits precision. It is used if we want to use memory effectively because it takes less memory in comparison to double data type. To define a float value, we must use a suffix f or F. Its default value is 0.0f. By default, float numbers are treated as double in Java.

- It occupies **4** bytes.
- It follows **single-precision** (6-7 decimal digits).
- There will be **no data loss** if we convert float to double.
- It uses **F** or **f** as a suffix. It is mandatory to add a suffix if you are declaring a float variable.

Range:

```
Range = 2^(bits - 1) - 1

For a float with 32 bits, the calculation would be:

Range = 2^(32 - 1) - 1 = 2^31 - 1 = 2147483647

Therefore, the range of a float in Java is approximately -2147483647 to 2147483647.
```

For example, if we define a float number as:

```
1. float height = 167.7
```

The above declaration of float variable gives the compilation error. We can correct the error by adding a suffix f or F.

```
1. float height = 167.7f
2. or
3. float height = 167.7F
```

Floating-Point Data Type	Values	Size (bits)*	Storage Requirement (bytes)	Default Value	Precision	Decimal Digits
float	IEEE 754 Floating-Point	32	4	0.0f	Single	6 decimal digits
double	IEEE 754 Floating-Point	64	8	0.0d	Double	15 decimal digits

double Data Type

The double data type is a 64-bit double-precision IEEE 754 floating-point number. It means that it gives 15-16 decimal digits precision. It consumes more memory in comparison to the float data type. It is used to store decimal values. Its default value is 0.0d. It is optional to add suffix d or D. For example:

1. **double** price = 987.90D
   (or)
2. **double** price = 987.90d
   (or)
3. **double** price = 987.90

- It follows **double-precision** (15-16 decimal digits).
- The **double** keyword is used to define a double-precision number.
- There will be **data loss** if we convert double to float.
- It uses **d** or **D** as a suffix. It is optional to add a suffix if you are declaring a double variable.

Range:
For double: Range = 2^(64 - 1) - 1 = 2^63 - 1 = 9,223,372,036,854,775,807The `double` data type uses 1 bit for the sign, 11 bits for the exponent, and 52 bits for the significand (mantissa). Similar to the `float` data type, the actual range and precision of `double` numbers can vary slightly due to the limitations of binary representation.