1. Compare and contrast the float and Decimal classes' benefits and drawbacks.

Answer: Both the float and decimal types store numerical values in Python.

We use floats when convenience and speed matter. A float gives us an approximation of the number we declare.

We use decimals when precision matters. Decimals can suffer from their own precision issues, but generally, decimals are more precise than floats. The performance difference between float and decimal, with Python 3, is not outlandish, and in my experience, the precision benefits of a decimal outweigh the performance benefits of a float.

2. Decimal('1.200') and Decimal('1.2') are two objects to consider. In what sense are these the same object? Are these just two ways of representing the exact same value, or do they correspond to different internal states?

Answer: Both values are same but internal representation at storage Is different. Precsion differs, Decimal('1.200') gives internally 1.200 and Decimal('1.2') gives 1.

3. What happens if the equality of Decimal('1.200') and Decimal('1.2') is checked?

Answer: Both values are checked to be equal since they only differ in precision.

4. Why is it preferable to start a Decimal object with a string rather than a floating-point value?

Answer: Decimal "is based on a floating-point model which was designed with people in mind, and necessarily has a paramount guiding principle – computers must provide an arithmetic that works in the same way as the arithmetic that people learn at school." – excerpt from the decimal arithmetic specification.

5. In an arithmetic phrase, how simple is it to combine Decimal objects with integers?

Answer: We can do it with use of Decimal().Decimal objects cannot generally be combined with floats or instances of fractions.

6. Can Decimal objects and floating-point values be combined easily?

Answer: No Decimal objects cannot generally be combined with floats or instances of fractions.

7. Using the Fraction class but not the Decimal class, give an example of a quantity that can be expressed with absolute precision.

Answer: Value of 0.5 will be represented as ½.

8. Describe a quantity that can be accurately expressed by the Decimal or Fraction classes but not by a floating-point value.

Answer. Decimal is more accurate than Float and Double.Float stores an approximate value and decimal stores an exact value. In summary, exact values like money should use decimal, and approximate values like scientific measurements should use float. When multiplying a non integer and dividing by that same number, decimals lose precision while floats do not.

Q9.Consider the following two fraction objects: Fraction(1, 2) and Fraction(1, 2). (5, 10). Is the internal state of these two objects the same? Why do you think that is?

Answer: Both will be reduced to one half

Q10. How do the Fraction class and the integer type (int) relate to each other? Containment or inheritance?

Answer: Fraction class and integer type(int) are related in form of a container. It contains two ints, one in the numerator and the other in the denominator

Colab paid products - Cancel contracts here