**Character Class in Java**

The Character class in Java is a **wrapper class** for the **char primitive data type**. It provides various **utility methods to manipulate and analyze characters**. Some key features of the Character class include:

* **Character Wrapping**: Converts char to Character objects and vice versa.
* **Character Checking**: Methods to check if a character is a letter, digit, whitespace, etc.
* **Case Conversion**: Methods to convert characters between uppercase and lowercase.
* **Unicode Support**: Handles Unicode character properties.

**Commonly Used Methods in Character Class**

1. isLetter(char ch): Checks if the character is a letter.
2. isDigit(char ch): Checks if the character is a digit.
3. isWhitespace(char ch): Checks if the character is a whitespace character.
4. isUpperCase(char ch): Checks if the character is uppercase.
5. isLowerCase(char ch): Checks if the character is lowercase.
6. toUpperCase(char ch): Converts a character to uppercase.
7. toLowerCase(char ch): Converts a character to lowercase.
8. isAlphabetic(int codePoint): Checks if the character is alphabetic.
9. isDefined(char ch): Checks if the character has a defined Unicode value.
10. getNumericValue(char ch): Returns the numeric value of a character.

**Coding**

**Basic Level Questions**

1. **Check if a given character is a letter**
   * **Input:** 'A'
   * **Output:** true
2. **Check if a given character is a digit**
   * **Input:** '5'
   * **Output:** true
3. **Check if a given character is an uppercase letter**
   * **Input:** 'G'
   * **Output:** true
4. **Check if a given character is a lowercase letter**
   * **Input:** 'g'
   * **Output:** true
5. **Convert a character to uppercase**
   * **Input:** 'b'
   * **Output:** 'B'
6. **Convert a character to lowercase**
   * **Input:** 'M'
   * **Output:** 'm'
7. **Check if a character is a whitespace character**
   * **Input:** ' '
   * **Output:** true
8. **Find the numeric value of a character**
   * **Input:** '7'
   * **Output:** 7
9. **Check if a character is an alphabetic character**
   * **Input:** '9'
   * **Output:** false
10. **Check if a character is a defined Unicode character**

* **Input:** 'A'
* **Output:** true

**Intermediate Level Questions**

1. **Check if a given character is a letter or digit**

* **Input:** '#'
* **Output:** false

1. **Find the Unicode code point of a character**

* **Input:** 'A'
* **Output:** 65

1. **Check if a character can start a Java identifier**

* **Input:** '1'
* **Output:** false

1. **Check if a character can be part of a Java identifier**

* **Input:** '$'
* **Output:** true

1. **Compare two characters lexicographically**

* **Input:** 'A', 'B'
* **Output:** -1

1. **Convert a digit to its corresponding character**

* **Input:** 5
* **Output:** '5'

1. **Find the category of a given character**

* **Input:** 'A'
* **Output:** UPPERCASE\_LETTER

1. **Check if a character is a control character**

* **Input:** '\n'
* **Output:** true

1. **Check if a character is a punctuation mark**

* **Input:** '.'
* **Output:** true

1. **Convert an integer to a character using radix**

* **Input:** (10, 16)
* **Output:** 'A'

**String Manipulation Using Character Class**

1. **Count the number of uppercase letters in a string**

* **Input:** "Hello World"
* **Output:** 2

1. **Count the number of lowercase letters in a string**

* **Input:** "Hello World"
* **Output:** 8

1. **Count the number of digits in a string**

* **Input:** "My age is 25"
* **Output:** 2

1. **Extract digits from a given string**

* **Input:** "Price: 500 USD"
* **Output:** "500"

1. **Convert all characters of a string to uppercase**

* **Input:** "hello"
* **Output:** "HELLO"

1. **Convert all characters of a string to lowercase**

* **Input:** "HELLO"
* **Output:** "hello"

1. **Remove all whitespace from a string**

* **Input:** "Hello World"
* **Output:** "HelloWorld"

1. **Reverse case of characters in a string**

* **Input:** "Hello"
* **Output:** "hELLO"

1. **Check if a string is a valid numeric string**

* **Input:** "12345"
* **Output:** true

1. **Find the first non-repeating character in a string**

* **Input:** "swiss"
* **Output:** 'w'

**Advanced Level Questions**

1. **Capitalize the first letter of every word in a string**

* **Input:** "hello world"
* **Output:** "Hello World"

1. **Find the frequency of each character in a string**

* **Input:** "banana"
* **Output:** {b=1, a=3, n=2}

1. **Remove all non-alphabetic characters from a string**

* **Input:** "H3llo, W0rld!"
* **Output:** "HlloWrld"

1. **Remove all non-digit characters from a string**

* **Input:** "H3llo, W0rld!"
* **Output:** "30"

1. **Check if two strings are anagrams using character counts**

* **Input:** "listen", "silent"
* **Output:** true

1. **Replace all vowels in a string with '\*'**

* **Input:** "hello"
* **Output:** "h\*ll\*"

1. **Check if a string contains only uppercase letters**

* **Input:** "HELLO"
* **Output:** true

1. **Check if a string contains only lowercase letters**

* **Input:** "hello"
* **Output:** true

1. **Sort characters in a string alphabetically**

* **Input:** "dcba"
* **Output:** "abcd"

1. **Find the most frequent character in a string**

* **Input:** "banana"
* **Output:** 'a'

**Edge Case & Debugging Questions**

1. **Check if a string contains only letters using Character class**

* **Input:** "Hello123"
* **Output:** false

1. **Remove duplicate characters from a string**

* **Input:** "banana"
* **Output:** "ban"

1. **Check if a string has balanced parentheses**

* **Input:** "(a+b)"
* **Output:** true

1. **Check if a string is a palindrome considering case sensitivity**

* **Input:** "Madam"
* **Output:** false

1. **Find the longest word in a sentence**

* **Input:** "Java is awesome"
* **Output:** "awesome"

1. **Replace all occurrences of a character in a string**

* **Input:** "apple", 'p', '\*'
* **Output:** "a\*\*le"

1. **Convert a sentence into Pig Latin**

* **Input:** "hello"
* **Output:** "ellohay"

1. **Reverse words in a sentence**

* **Input:** "Hello World"
* **Output:** "World Hello"

1. **Check if a character is a valid hexadecimal digit**

* **Input:** 'F'
* **Output:** true

1. **Replace every second character in a string with \***

* **Input:** "hello"
* **Output:** "h\*l\*o"