22 . Write a High-level code that can perform a letter frequency attack on an additive cipher without human intervention.

**CODE :**

**import string**

**def letter\_frequency\_attack(ciphertext):**

**letter\_frequencies = {}**

**total\_letters = 0**

**for letter in ciphertext:**

**if letter.isalpha():**

**letter = letter.lower()**

**letter\_frequencies[letter] = letter\_frequencies.get(letter, 0) + 1**

**total\_letters += 1**

**for letter, frequency in letter\_frequencies.items():**

**letter\_frequencies[letter] = frequency / total\_letters**

**sorted\_frequencies = sorted(letter\_frequencies.items(), key=lambda x: x[1], reverse=True)**

**most\_frequent\_letter = sorted\_frequencies[0][0]**

**key = (ord(most\_frequent\_letter) - ord('e')) % 26**

**plaintext = ""**

**for letter in ciphertext:**

**if letter.isalpha():**

**if letter.isupper():**

**decrypted\_letter = chr((ord(letter) - ord('A') - key) % 26 + ord('A'))**

**else:**

**decrypted\_letter = chr((ord(letter) - ord('a') - key) % 26 + ord('a'))**

**plaintext += decrypted\_letter**

**else:**

**plaintext += letter**

**return key, plaintext**

**ciphertext = "Bzdrzq bhogdq? H oqdedq Bzdrzq rhq ybdr?"**

**key, plaintext = letter\_frequency\_attack(ciphertext)**

**print("Key:", key)**

**print("Plaintext:", plaintext)**

**OUTPUT :**

