1. Adding new natural languages to the text adventure game is more closely with the dimension of “Adding new types with existing behaviors”.

type Language = "EN" | "SP" | "JA";

In the text adventure game, adding support for new languages does involve introducing a new type (Language) to work with existing game behavior. We also will create new function to select the language based on user input and to perform translations.

1. Based on your answer to question 1, describe a plan for refactoring the text adventure code so that it naturally supports the kind of extensibility that is relevant to your task.

-Define the language type that includes all the supported languages.

-Create a function to handle language selection based on user input.

*export function selectLanguage(): Language {*

*//Implementation for language selection*

*},*

-Implement a translations object to store the language specific content.

*const translations: Record<Language, Record<string, string>> = {*

*EN: {*

*// English translations...*

*},*

*SP: {*

*// Spanish translations...*

*},*

*JA: {*

*// Japanese translations...*

*},*

*// Add more languages and translations as needed*

*};*

-Implement the game loop and room interactions using the translation object