# Tech Career Advisor Application - Code Documentation

## Skills\_Assessment.py

### Imported Libraries

* **streamlit (st)**: A web application framework for creating interactive web applications with Python.
* **pandas (pd)**: Data manipulation and analysis library that provides data structures like DataFrames.
* **sklearn.preprocessing.StandardScaler**: Standardizes features by removing the mean and scaling to unit variance.
* **sklearn.model\_selection.train\_test\_split**: Splits datasets into random train and test subsets.
* **sklearn.svm.SVC**: Support Vector Classification, a machine learning model for classification tasks.

### Code Structure and Flow

1. **Page Configuration**:
   * Sets up the Streamlit page with title, icon, and layout.
   * Adds a "Back to Home" button that navigates to the homepage.
   * Applies custom CSS for styling the assessment form.
2. **Data Setup**:
   * Creates a mapping dictionary that connects numerical prediction outputs to career roles.
   * Defines 15 questions related to various technical skills.
   * Sets up answer options from "Not Interested" to "Professional".
3. **Helper Functions**:
   * reset\_form(): Clears form entries after submission.
   * load\_and\_train\_model(): Loads training data from CSV, preprocesses it, and trains an SVC model.
   * mapping(): Converts text responses to numerical values for model processing.
4. **Main Function**:
   * Displays form header and introduction.
   * Loads the machine learning model if not already in session state.
   * Creates a form with all 15 questions as radio buttons.
   * Processes form submission:
     + Maps responses to numerical values
     + Uses the model to predict a career path
     + Displays the predicted role
     + Shows technical details in an expandable section
5. **Execution**:
   * Uses the standard Python if \_\_name\_\_ == "\_\_main\_\_" pattern to run the main function when the script is executed directly.

### Key Features

* Uses a trained Support Vector Classifier to recommend a tech career path.
* Provides a clean, responsive UI with styled form elements.
* Preserves model state between interactions using Streamlit's session state.
* Shows detailed information about the prediction process for transparency.

## Homepage.py

### Imported Libraries

* **streamlit (st)**: Web application framework for creating the interface.

### Code Structure and Flow

1. **Page Configuration**:
   * Sets up the Streamlit page with title, icon, and layout.
   * Applies custom CSS for styling the homepage elements.
2. **Header Section**:
   * Displays the title "Tech Career Advisor" and subtitle.
   * Shows a career path illustration (loaded from an external URL).
   * Presents an introductory paragraph about the app's purpose.
3. **How It Works Section**:
   * Uses a three-column layout to explain the assessment process:
     + Complete the assessment
     + AI analysis
     + Get career match
4. **Call to Action**:
   * Provides a prominent button to navigate to the Skills Assessment page.
5. **Career Information Section**:
   * Displays a grid of expandable sections for different tech careers.
   * Each expander contains a description of a specific tech role.
   * Organized in a three-column layout for better readability.
6. **Footer**:
   * Displays copyright information and credits.

### Key Features

* Clean, visually appealing layout with consistent styling.
* Responsive design elements that adapt to different screen sizes.
* Interactive elements like expandable career information cards.
* Clear navigation path to the assessment tool.

## Application Flow

1. Users start at the homepage (Homepage.py).
2. Homepage explains the purpose of the app and how the assessment works.
3. Users click "Take the Assessment Now" to navigate to the Skills Assessment page.
4. On the Skills Assessment page, users answer 15 questions about their technical skills.
5. Upon submission, the app processes their responses through a machine learning model.
6. The app displays a predicted career path based on the user's responses.
7. Users can return to the homepage using the "Back to Home" button.