1. Hello World: Create a new Python file, then get it to print “Hello World!” in the command prompt

2. Calculator: Create a new Python file, then define functions for addition, subtraction, multiplication, and division. Prompt the user to enter two numbers and an operator, then call the appropriate function based on the operator and print the result.

3. Hangman: Create a new Python file, then choose a word and display the number of letters in the word as underscores. Prompt the user to guess a letter, then replace any underscores in the word with the guessed letter if it appears in the word. Keep track of incorrect guesses and display a hangman drawing as the number of incorrect guesses increases. End the game when the user either guesses the word or runs out of guesses.

4. Rock, Paper, Scissors: Create a new Python file, then prompt the user to input "rock", "paper", or "scissors". Generate a random choice for the computer, then compare the user's choice to the computer's choice and determine the winner based on the game rules. Print the result.

5. Web Scraper: Choose a website to scrape and find the HTML elements that contain the information you want to extract. Use a Python library such as BeautifulSoup or Scrapy to parse the HTML and extract the relevant data. Save the data to a file or print it to the console.

6. Weather App: Choose a weather API such as OpenWeatherMap or Weather Underground. Sign up for an API key and use it to make requests to the API using Python's requests library. Parse the response and extract the relevant weather information for the user's

input location. Display the weather information to the user.

7. Sentiment Analysis: Choose a dataset of text data such as reviews or tweets. Preprocess the data by cleaning and tokenizing it. Use a Python library such as NLTK or spacy to train a machine learning model on the data. Test the model on a separate test set and evaluate its accuracy. Use the model to predict the sentiment of new text data.

8. Chatbot: Choose a platform to build the chatbot such as Facebook Messenger, Slack, or Telegram. Create a new bot account and configure it to receive messages from users. Use a Python library such as ChatterBot or Dialogflow to build the chatbot's conversational logic. Train the chatbot on sample conversations and test it with real users.

9. Image Recognition: Choose a dataset of images and their corresponding labels such as the CIFAR-10 or ImageNet datasets. Use a Python library such as Keras or PyTorch to train a convolutional neural network on the data. Test the model on a separate test set and evaluate its accuracy. Use the model to predict the labels of new images.

10. Game AI: Choose a game to build the AI for such as chess or checkers. Define the game rules and create a Python program to simulate the game. Use a Python library such as TensorFlow or PyTorch to train a machine learning model on sample games. Test the model on a separate test set and evaluate its performance. Use the model to make intelligent moves in the game based on the current state.