

# **TITLE: EARTHQUAKE PREDICTION MODEL USING PYTHON.**

## **PROBLEM DEFINITION:**

\* Measuring energy consumption directly in Java typically requires specialized hardware or system-level access that is not readily available through standard Java libraries. However, you can use Java to interact with hardware or retrieve energy-related data if the necessary APIs or libraries are available for your specific platform and hardware.

## **DESIGN:**

### **1. PROGRAM DESCRIPTION:**

\* The program simulates earthquake predictions based on random magnitudes.

### **2. FUNCTION DEFINITION:**

``predict_earthquake()`:`

- \* This function generates a random earthquake prediction based on magnitude.

- \* Input: None.

- \* Output:

A string representing the prediction ("No Earthquake," "Minor Earthquake," "ModerateEarthquake," or "Major Earthquake").

### **3.MAIN PROGRAM:**

- Within the ``if __name_`

`"__main__": block`

Call the

``predict_earthquake()```

function.

- \* Display the generated earthquake prediction.

## **4.ALGORITHM :**

Import the random module.

\* Define the

`predict\_earthquake function:

\* Generate a random magnitude between 2.0 and 9.0.

\* Check the magnitude to classify the prediction.

Output the generated earthquake prediction.

## **5. EXAMPLE OUTPUT:**

An example of what the program's output might look like:

Earthquake Prediction: Moderate Earthquake

## **6.ADDITIONAL CONSIDERATION:**

\* This is a simple simulation for educational purposes and should not be used for actual earthquake prediction.

\* The program uses random values to generate predictions, which will vary with each run.

Real earthquake prediction models involve complex scientific methods, data analysis, and extensive research in seismology.

\* This textual design outlines the key components and logical flow of the program. You can use this design to implement the Python code based on the earlier provided program.

### **PROGRAM:**

```
import random
```

```
def predict_earthquake():
```

```
    # Simulate a basic random prediction
```

```
    magnitude = random.uniform(2.0, 9.0)
```

```
    if magnitude < 5.0:
```

```
        return "No Earthquake"
```

```
    elif magnitude < 6.0:
```

```
        return "Minor Earthquake"
    elif magnitude < 7.0:
        return "Moderate Earthquake"
    else:
        return "Major Earthquake"

if __name__ == "__main__":
    prediction = predict_earthquake()
    print(f"Earthquake Prediction: {prediction}")
```

## **OUTPUT:**

\* Earthquake Prediction: Moderate Earthquake