|  |  |
| --- | --- |
| Name Of The Student | Vaishnavi G |
| Internship Project Topic | Build a Classification Model for Drug Trials Dataset |
| Name of the Organization | TCS iON |
| Name of the Industry Mentor | Himdweep Walia |
| Name of the Institute | SRM Institute of Science and Technology |

|  |  |  |
| --- | --- | --- |
| Date | Day # | Hours Spent |
| 19/11/2022 | 39 | 5 hours |
| Activities done during the day:  **Project Hands on - Check effectiveness of drug and value counts**  **To check the effectiveness of the drugs**   |  | | --- | | df['effectiveness'].value\_counts().plot(kind='pie')  plt.show() |   **df -** A Data frame is a two-dimensional data structure, i.e., data is aligned in a tabular fashion in rows and columns.  **Value counts() -**  pandas is an open-source Python library that provides operations to analyze and manipulate data structures called data frames. The value\_counts() function in pandas returns a series that contains the number of unique values. A series is a one-dimensional array.  **Parameters**  **The value\_counts function takes the following parameters:**   * **normalize (optional):** If set to True, the function returns the relative frequencies of the values. It is set to False by default. * **sort (optional):** If set to True, the function returns the values in a sorted manner. It is set to True by default. * **ascending (optional):** If set to True, values are sorted in an ascending manner. It is set to False by default. * **bins (optional):** Groups values into bins instead of counting them. It is set to None by default. * **dropna (optional):** If set to True, counts of NaN are not included. It is set to True by default.   **Return value:**  The function returns a series of counts of unique values.  **plot() :**  Pandas uses the plot() method to create diagrams. We can use Pyplot, a submodule of the Matplotlib library to visualize the diagram on the screen.  **Scatter:**   * a scatter plot with the kind argument: * kind = 'scatter' * A scatter plot needs an x- and a y-axis. * Include the x and y arguments like this: x = 'Duration', y = 'Calories'   **Histogram :**   * Use the kind argument to specify that you want a histogram: * kind = 'hist' * A histogram needs only one column. * A histogram shows us the frequency of each interval   **Code :**   |  | | --- | |  |   **Output:**   |  | | --- | |  | | | |