**Building a Personalized Product Recommendation System based on User Preferences** using Flask.

**Problem Statement:**

**Develop a Flask application that recommends similar products based on user preferences for price, number of favorites, and number of reviews.**

This could be used by users looking for personalized product recommendations when shopping for baby products or gifts. You could provide recommendations based on criteria like budget, popularity, or high reviews.

Solution Plan :

* **Price**: Users may want products within a specific price range.
* **Favorites**: Popularity based on how many people favorited the product.
* **Number of Reviews**: Customer feedback can influence product recommendations.

The goal is to build an application where users input their preferences (e.g., price range, minimum reviews, popularity), and the system returns similar items from the dataset.

How we will do it ?

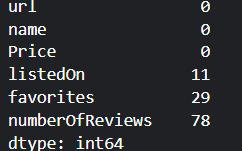
Step 1 : I m taking etsy baby items data set

Step 2 : importing the libraries.

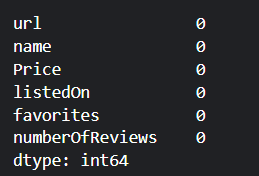
Step 3: Reading the dataset

Step 4 : Converted listed\_on to date and time format like firstly it was like and I made it to this . Only purpose of doing this was making the dataset more readable format so that it doesn’t throw error lately.

Step 5: Data cleaning and preoaration finding the missing values if your have some missing values caught and immediately fix this using .fillna function.



After cleaning we got



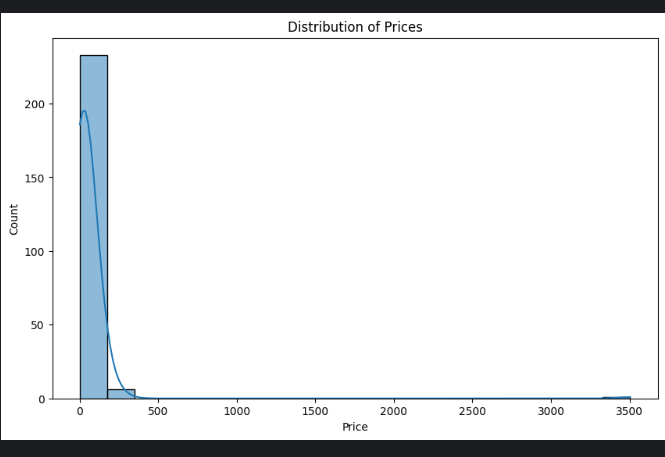
Step 6 : Performing Exploratory Data Analysis (EDA) using df.describe()

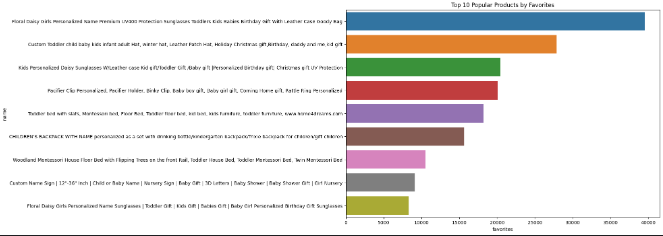
Step 7 : Performing Feature Engineering like extracting the numerical values and categorical values , numerical values like prices contains + $ signs so we need to fix this as converting this into str .astype and then replacinging .str.replace

Step 8 : Performing label encoding which will be helpful when we have been using correlation heatmap.

Step 9 : Then we extracted components like year and month from the listedOn list using .dt accessor.

Step 10 : Then we performed some visulizations like distribution of prices ,

 listing out top 10 popular products on favourites ,



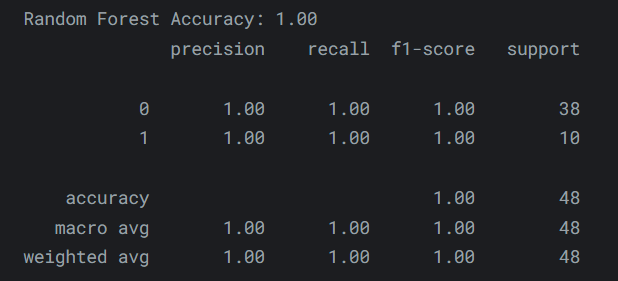
Step 11: Listing out a HeatMap a correlation heatmap for all numeric values plotted.

Step 12 : performing scatterplot for prices and favourites

Step 13 : Implementing some machine learning models , since we need an answer like based on the user input we will be using a classification supervised machine learning algorithm.

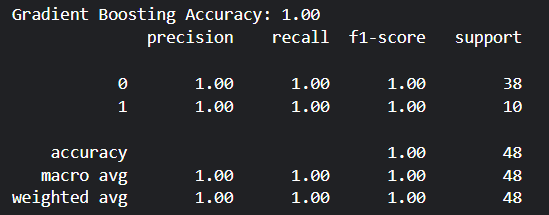
Step 14 : Before this train test and split the data you do this when x will be your feature variable and y will be target variable .

Step 15 : Implementing your Random forest classifier which reduces risk of overfitting implement and giving us an accuracy score 0f 1 which means the model is perfect and the data is evenly giving us right answer when we predicted some values also we added some extra metrices like precision , recall , f1- score etc.

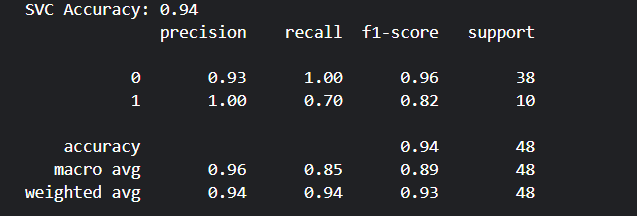


Step 16 : then we implemented KNN model but since the model was was giving us an accuracy of 0.03 which means very poor model so indeed we ask for some recommendation but still didn’t get anything.

Step 17 : For this model being a poor performer we added a gradient boost classifier giving an accuracy of 1, it mostly used for improving the performance and reducing bias risk.

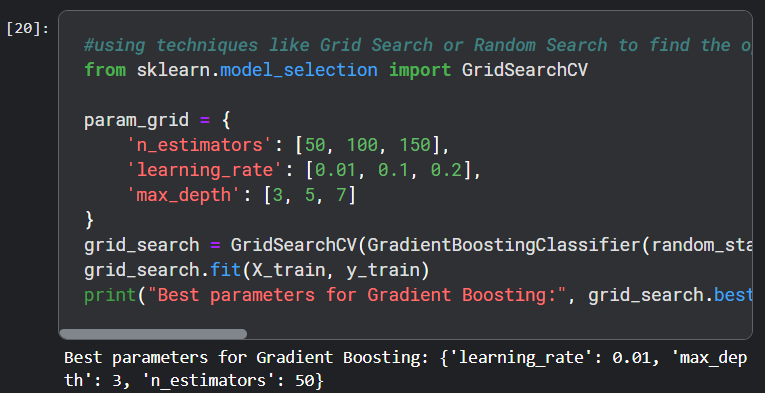


Step 18 : Then we implemented Support Vector machine in which we got an accuracy of 0.94 which was not good .

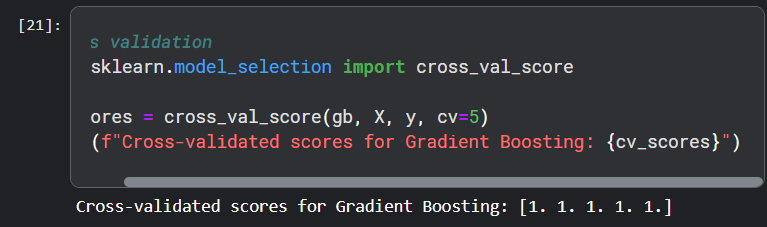


Step 19 : we implemented some techniques like gried search and random serach to find the best parameter .

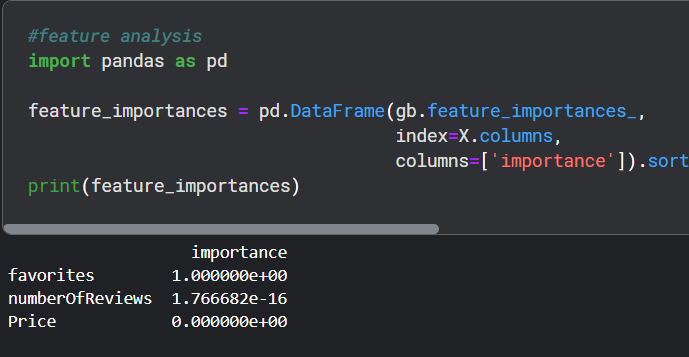
The only requirement of grid search is that it tries every combination in a grid once (and only once). For random search, we input the domain and each time the algorithm gives us a random combination of hyperparameter values to try.



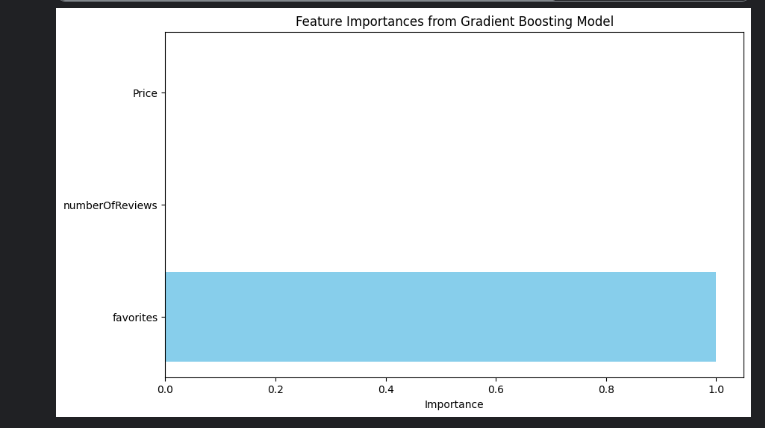
Step 20 : Cross validation score , next we had cross validation score in which we had the acuuracy score for each datapoints passing through gradient boosting.



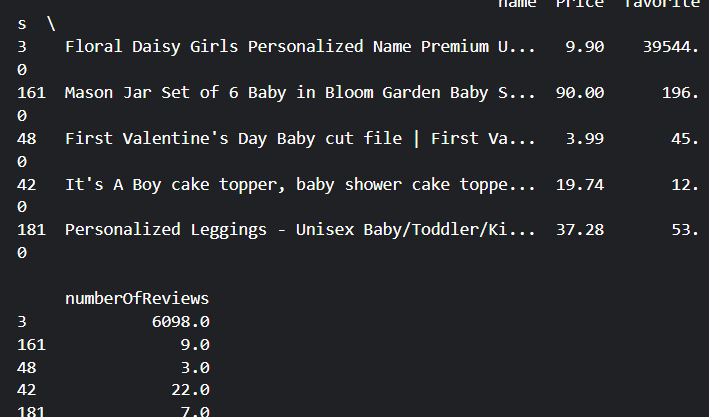
Step 21: Next we performed feature analysis on the columsn like favourites , numberoffavourites , and price .



Step 22 : performing a visulaization about the feature analysis we got to know ,

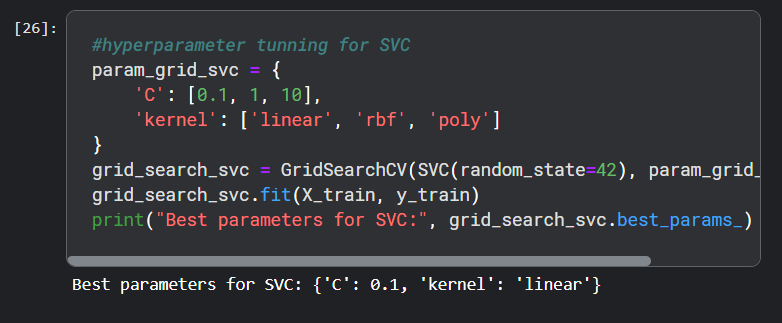


Step 23 : we implemented KNN model and we got some recommendations with random input we gave .

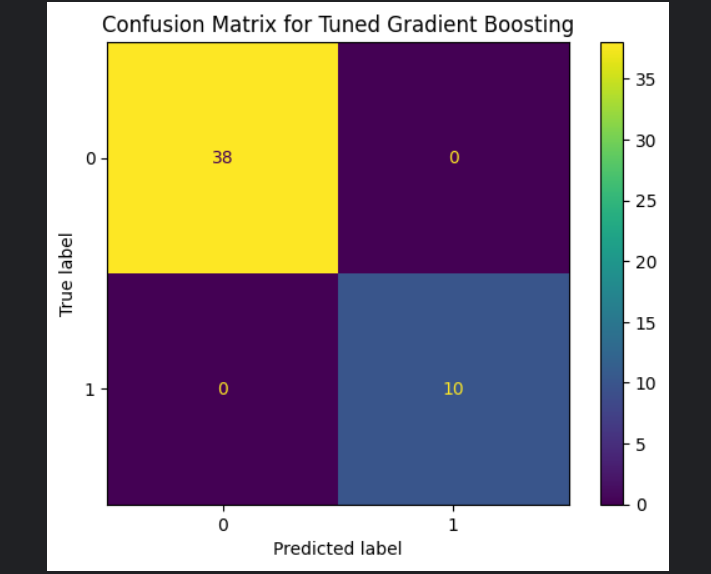


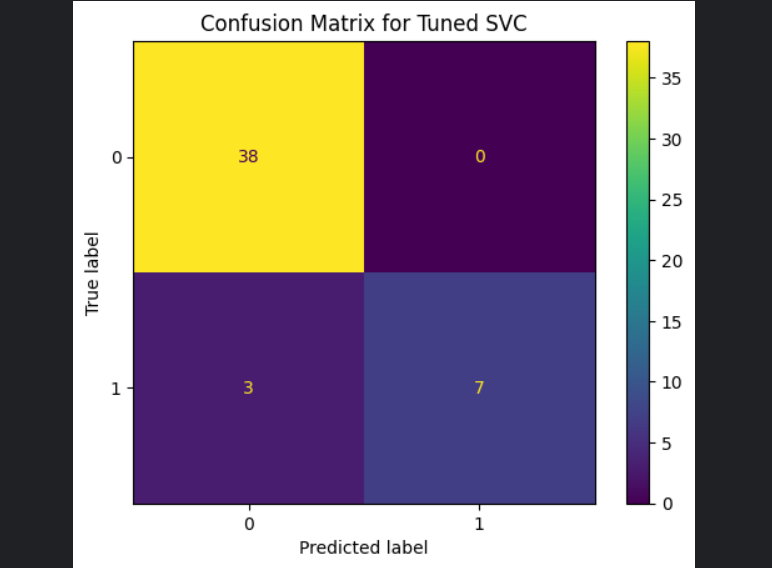
Step 24 : we evaluated the tuned model in which we used grid Serach and imported our SVM model

Hyperparameter tuning is an essential part of the machine learning process, as it allows you to fine-tune the performance of your model. You can use methods like grid search, random search, and bayesian optimization to tune hyperparameters



Step 25 : we visualized the data using some performance metrices like confusion matrix , aand tune gradient boosting and other visualization was confusion matrix and tuned SVC





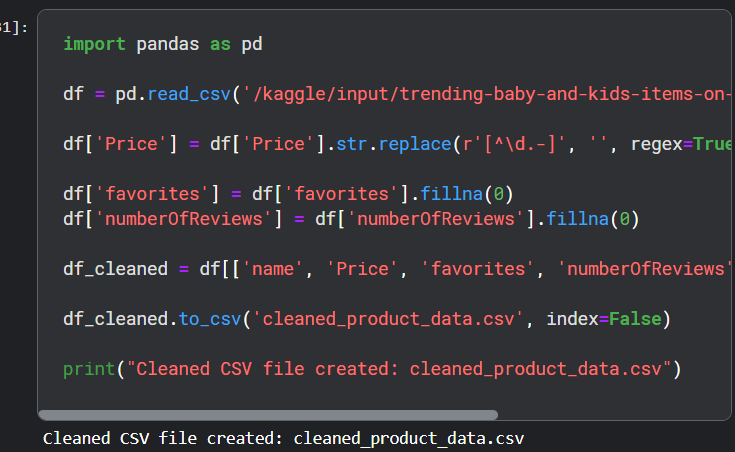
Conclusions Drawn :

**Random Forest and Gradient Boosting** achieved perfect accuracy.

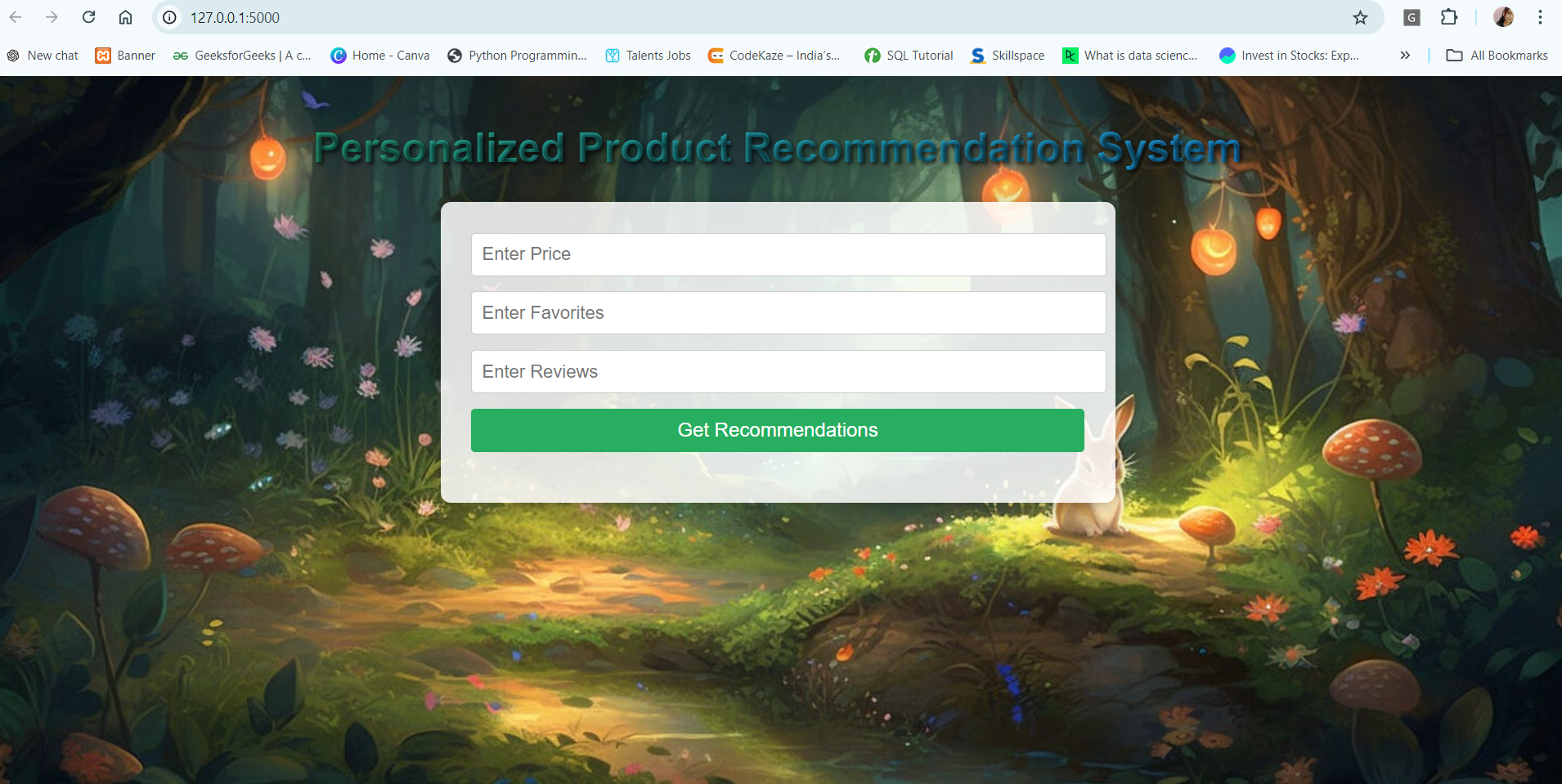
**SVC** had slightly lower performance but still significant.

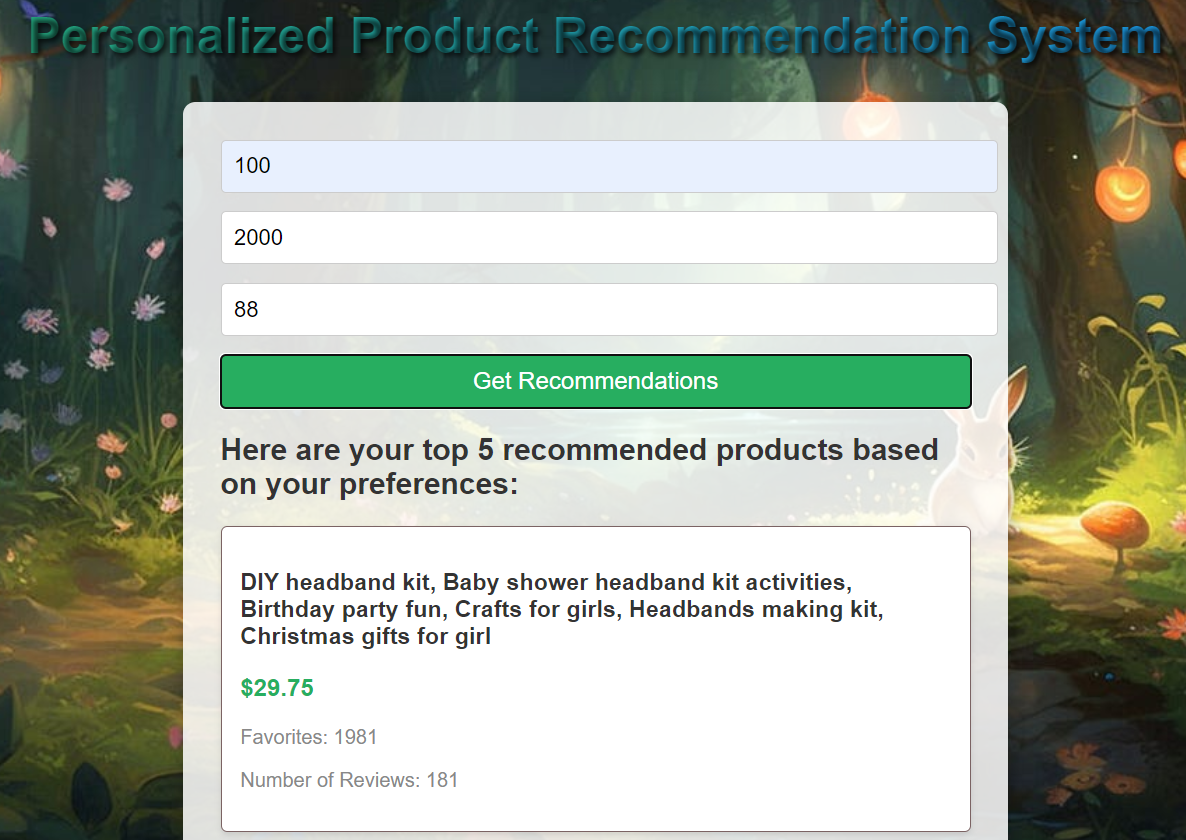
**Feature Importance** shows that favorites is the most influential feature.

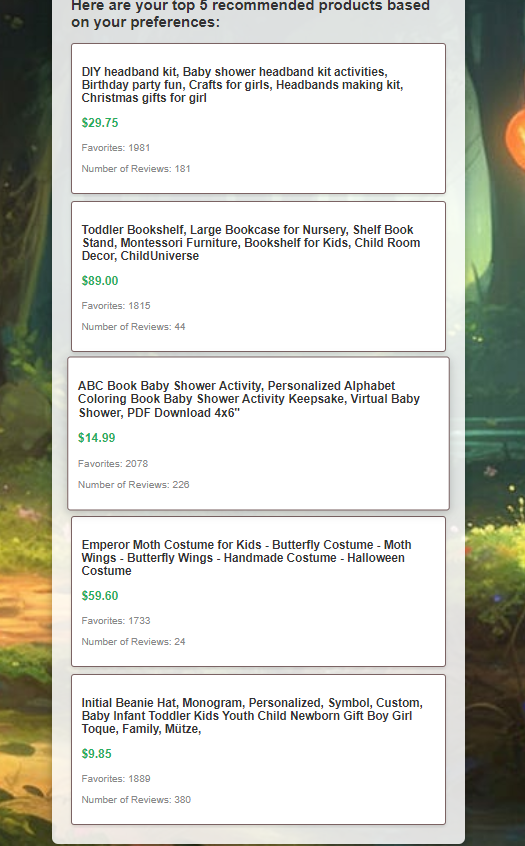
Step 26 : Exporting the trained model .pkl files , I will be develoving flask application that too I will be deploying it on a local server so I will be required .csv filed extracted one .



My output :







We got five columns in which we can now buy our product its listed according to decreeing order.

# Developing a flask application .

Step 1 : make a new folder named flask

Step 2 : download and keep the cleaned\_dataset.csv file and add it to this flask folder

Step 3 : open code editor I have VS code so I will be opening that

Make a file name app.py , make a folder name templates which will store all your .html files and static will store all your images and css file.

Structure should be like

Flask -|

|- app.py

|- Cleaned\_data.csv

|- Templates -|

|- index.html

|- recommendations.html

|- Static -|

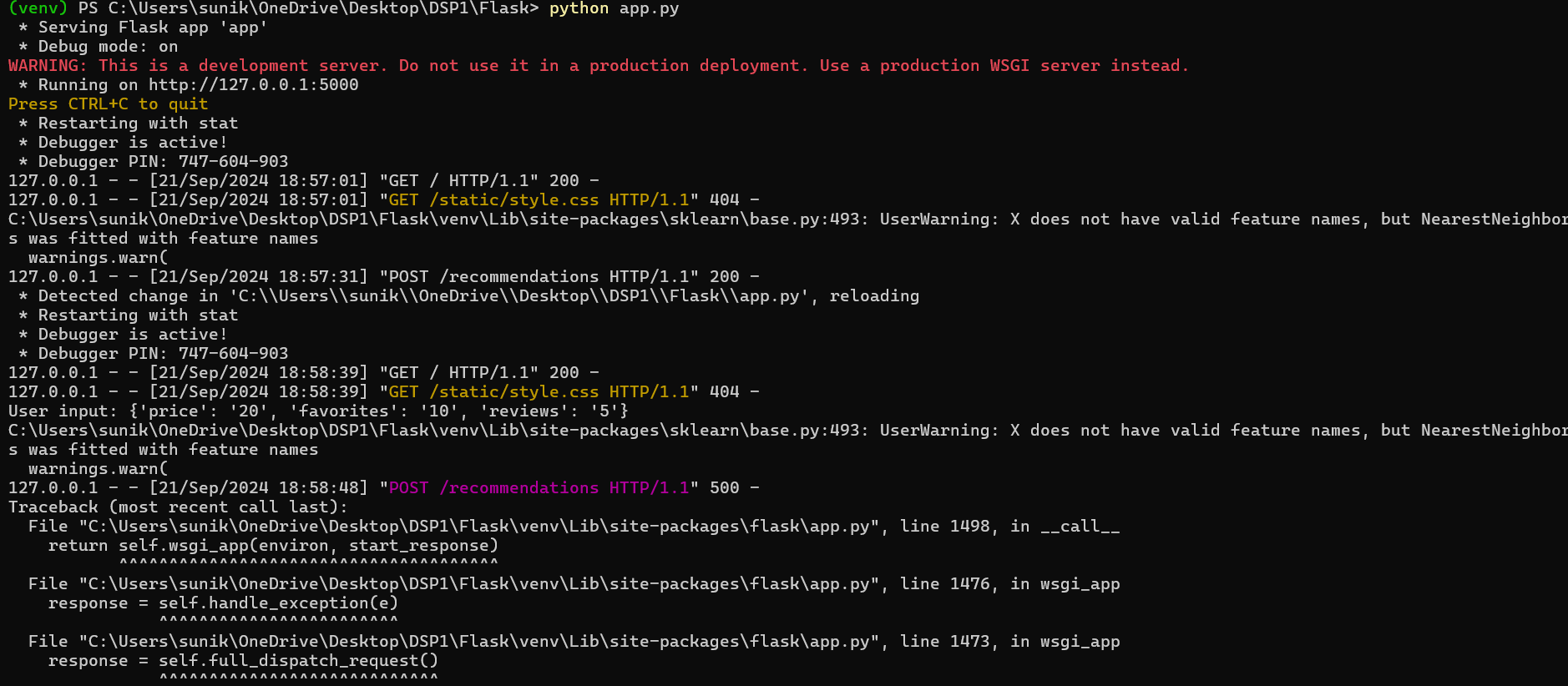
|- background.jpg

Step 4 : some the code you can find the flask folder on my github.

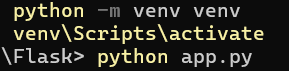
Step 5 : specify a requirements.txt file in which 4 packages needs to be installed in your system.

Step 6 : For deploying your model in the local web browswer you will be needed to install this

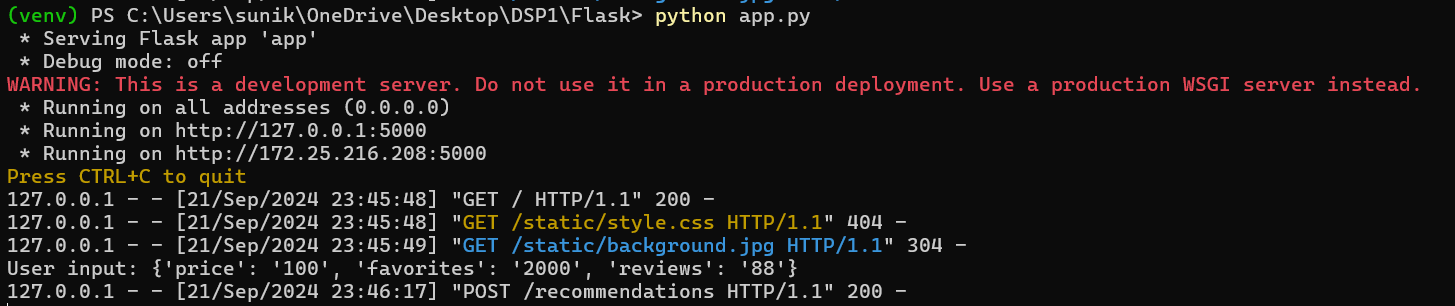


Maybe you can face some issues realted to module keep trying install correct version.

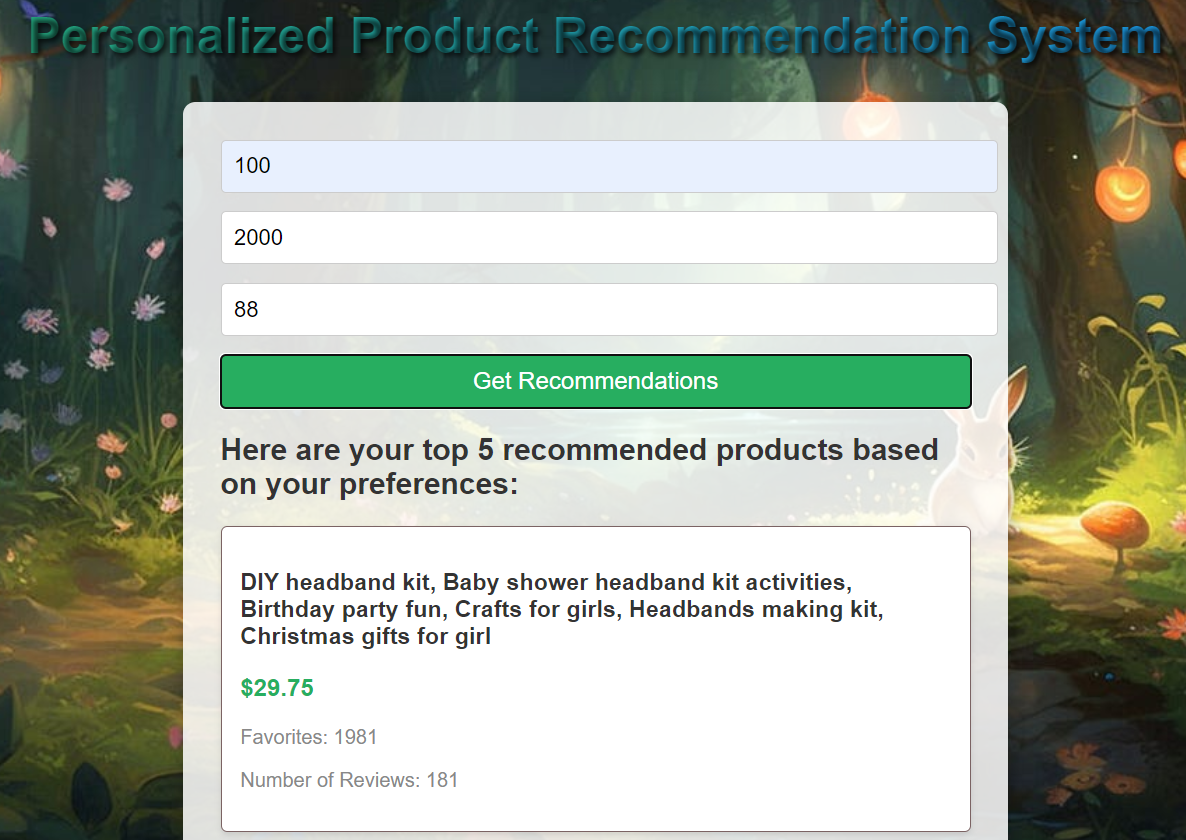
Step 7 : After the packages are successfully installed make sure you run these life for compliation

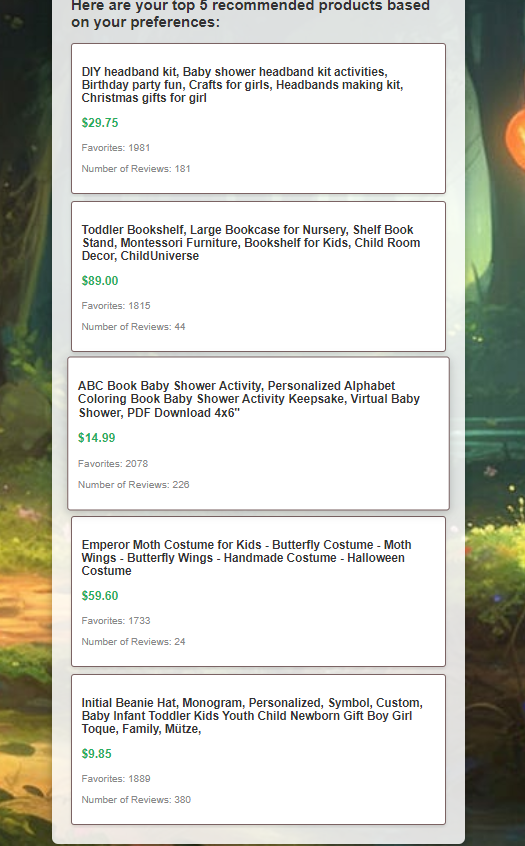


Step 8 : Once the .Venv script is activated you will see that you are able to run a program and here you will be see a server location as <HTTP://127.0.0.1:5000> click click and boom your ouput is here :



# My Recommender system Output It look like and how it work:





\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_END\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_