**PYTHON**

**Question: 1**

**You have an input dictionary given,**

input\_dict = {"abc":{"def":{"ghi":{"jkl":{"mno":{"pqr":{"stu":{"vwx":{"yz":"you are finally here !!!"}}}}}}}}}

**Task: You have to write a Python function that will take this input and print it like that,**

output = {"abc":["def","ghi","jkl","mno","pqr","stu","vwx","yz"],

"def":["ghi","jkl","mno","pqr","stu","vwx","yz"],

"ghi":["jkl","mno","pqr","stu","vwx","yz"],

"jkl":["mno","pqr","stu","vwx","yz"],

"mno":["pqr","stu","vwx","yz"],

"pqr":["stu","vwx","yz"],

"stu":["vwx","yz"],

"vwx":["yz"],

"yz":["you are finally here !!!"]}

**ANS. Check the repository notebook file ->** [**soodsid/Test-3-10-2024 (github.com)**](https://github.com/soodsid/Test-3-10-2024)

**Question: 2**

**Given an array of length ‘N’, where each element denotes the position of a stall. Now you have ‘N’ stalls and an integer ‘K’ which denotes the number of horses that are mad. To prevent the horses from hurting each other, you need to assign the horses to the stalls, such that the minimum distance between any two of them is as large as possible. Return the largest minimum distance.**

**array: 1,2,4,8,9 & k=3**

**O/P: 3**

**Explanation: 1st horse at stall 1, 2nd horse at stall 4 and 3rd horse at stall 8**

**ANS. Check the repository notebook file ->** [**soodsid/Test-3-10-2024 (github.com)**](https://github.com/soodsid/Test-3-10-2024)

**Question: 3**

****

**Mr. Karthiken works in a door mat manufacturing company. One day, he designed a new door mat with the following specifications:**

**a) Mat size must be N X M. (N is an odd natural number, and M is 3 times N.)**

**b) The design should have ‘WELCOME’ written in the center.**

**c) The design pattern should only use |, . and – characters.**

**Sample Design is given above image, Write a python code for this.**

**ANS. Check the repository notebook file ->** [**soodsid/Test-3-10-2024 (github.com)**](https://github.com/soodsid/Test-3-10-2024)

**Question: 4**

**Given an array nums of n integers, return an array of all the unique quadruplets [nums[a], nums[b], nums[c], nums[d]] such that:**

**a) 0 <= a, b, c, d < n**

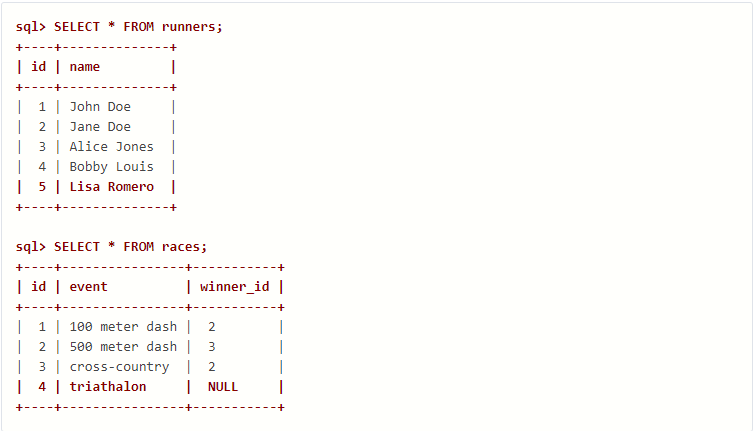
**b) a, b, c, and d are distinct.**

**c) nums[a] + nums[b] + nums[c] + nums[d] == target**

**ANS. Check the repository notebook file ->** [**soodsid/Test-3-10-2024 (github.com)**](https://github.com/soodsid/Test-3-10-2024)

**SQL:**

**Question: 1**

**Given the following tables:**

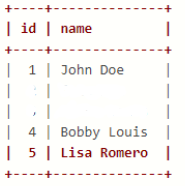
**What will be the result of the query below?**

SELECT \* FROM runners WHERE id NOT IN (SELECT winner\_id FROM races)

**Explain your answer and also provide an alternative version of this query that will avoid the issue that it exposes.**

**ANS.**

**This query will return all the runners who have never won a race.**

****

**id name**

**1 John Doe**

**4 Bobby Louis**

**5 Lisa Romero**

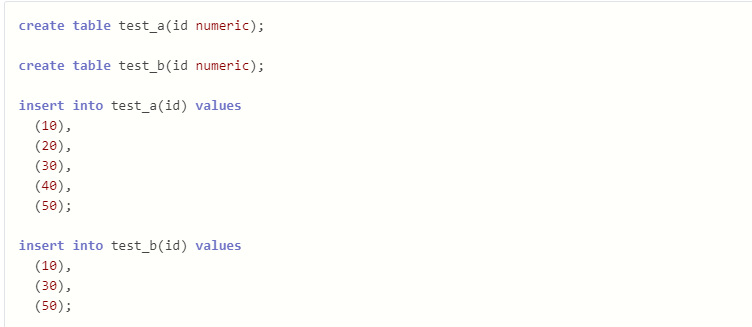
**But this query may show unexpected behaviour as we can see that our races table have a null value.**

**Alternative solution:**

**SELECT \* FROM runners WHERE id NOT IN (SELECT winner\_id FROM races where winner\_id is not NULL)**

**Question: 2**

**Given two tables created as follows**

****

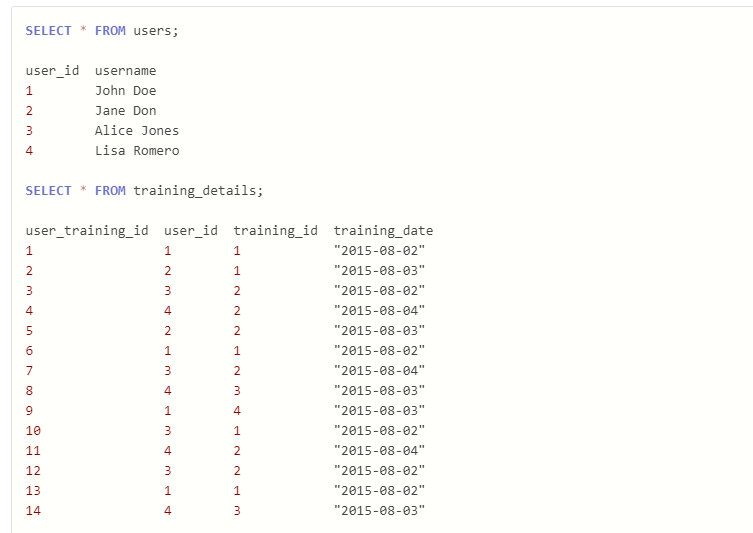
**Write a query to fetch values in table test\_a that are and not in test\_b without using the NOT keyword.**

**ANS.**

**SELECT a.id FROM test\_a a LEFT JOIN test\_b b ON a.id = b.id WHERE b.id IS NULL;**

**Question: 3**

**Given the following tables:**

****

**Write a query to to get the list of users who took the a training lesson more than once in the same day, grouped by user and training lesson, each ordered from the most recent lesson date to oldest date.**

**Ans.**

**SELECT**

**u.user\_id,**

**u.username,**

**td.training\_id,**

**td.training\_date,**

**COUNT(\*) AS num\_lessons\_taken**

**FROM**

**users u**

**JOIN**

**training\_details td ON u.user\_id = td.user\_id**

**GROUP BY**

**u.user\_id,**

**u.username,**

**td.training\_id,**

**td.training\_date**

**HAVING**

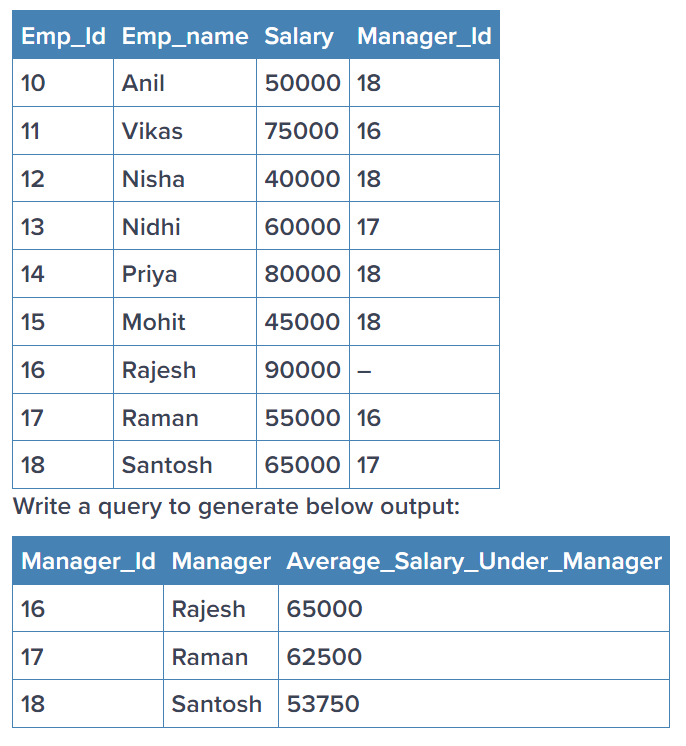
**COUNT(\*) > 1**

**ORDER BY**

**td.training\_date DESC;**

**Question: 4**

**Consider the Employee table below.**

****

**ANS.**

**SELECT**

**t.Manager\_id,**

**(SELECT**

**CASE**

**WHEN Manager\_id = '-' THEN 'Rajesh'**

**ELSE Emp\_name**

**END**

**FROM employee**

**WHERE Emp\_id = t.Manager\_id) AS Manager,**

**AVG(t.Salary) AS Average\_Salary\_Under\_Manager**

**FROM**

**employee t**

**WHERE**

**t.Manager\_id != '-'**

**GROUP BY**

**t.Manager\_id;**

**STATS**

**Question: 1**

What is the meaning of six sigma in statistics? Give proper example

**ANS.**

Six Sigma is a statistical methodology aimed at improving process quality by minimizing defects. It involves a structured approach of Define, Measure, Analyze, Improve, and Control (DMAIC). The goal is to achieve near-perfect performance with only 3.4 defects per million opportunities. For example, in manufacturing, Six Sigma can

be used to reduce defects in processes such as welding by identifying root causes and implementing improvements to reduce variation and defects.

An example of Six Sigma in action can be seen in manufacturing processes. Let's say a company produces automobile components, and one of their critical processes involves welding parts together. However, they've been experiencing defects such as weld cracks, uneven welds, or welds that don't meet strength requirements.

To apply Six Sigma:

Define: The company defines the problem as high defect rates in the welding process, which result in increased rework and customer complaints.

Measure: Data is collected on defect rates, process cycle times, and other relevant metrics.

Analyze: Statistical analysis is conducted to identify factors contributing to weld defects, such as improper welding parameters, inconsistent material quality, or operator errors.

Improve: Changes are made to the welding process, including optimizing welding parameters, implementing better quality control measures, and providing additional training to operators.

Control: Control mechanisms are put in place to monitor the welding process continuously. Regular audits, quality checks, and ongoing training ensure that the improvements are sustained over time.

By following the Six Sigma methodology, the company aims to reduce defect rates to within Six Sigma standards, resulting in higher-quality products, reduced waste, and increased customer satisfaction.

**Question: 2**

What type of data does not have a log-normal distribution or a Gaussian distribution? Give proper example

**ANS.**

Data that does not have a log-normal or Gaussian distribution includes count data, which often follows a Poisson or negative binomial distribution. Count data represents the number of occurrences of an event within a fixed interval, such as the number of customers arriving at a store or the number of defects in a product. These distributions are used to model scenarios where the data is skewed or non-symmetric, with the Poisson distribution describing the probability of a fixed number of events occurring and the negative binomial distribution allowing for over-dispersion when the variance is larger than the mean.

For example, consider a scenario where we are interested in the number of accidents occurring at a particular intersection in a day. This count data may follow a Poisson distribution if the rate of accidents per day remains relatively constant over time. However, if the intersection experiences periods of higher or lower traffic flow, leading to varying accident rates, the count data may better fit a negative binomial distribution due to the over-dispersion in the data.

**Question: 3**

What is the meaning of the five-number summary in Statistics? Give proper example

**ANS.**

The five-number summary in statistics provides a concise summary of the distribution of a dataset. It consists of five key values: the minimum, the first quartile (Q1), the median (Q2), the third quartile (Q3), and the maximum.

Minimum: The smallest value in the dataset.

First Quartile (Q1): The value below which 25% of the data fall. It is also the median of the lower half of the dataset.

Median (Q2): The middle value of the dataset when it is sorted in ascending order. It divides the dataset into two halves, with 50% of the data falling below it and 50% above it.

Third Quartile (Q3): The value below which 75% of the data fall. It is also the median of the upper half of the dataset.

Maximum: The largest value in the dataset.

The five-number summary is particularly useful for summarizing the central tendency and spread of a dataset, as well as identifying potential outliers.

**Question: 4**

What is correlation? Give an example with a dataset & graphical representation on jupyter Notebook

**ANS.**

Correlation is a statistical measure that describes the degree of association or relationship between two variables. It indicates whether and how strongly the variables change together. Correlation values range from -1 to 1:

1: Perfect positive correlation (as one variable increases, the other variable increases proportionally).

0: No correlation (the variables are independent of each other).

-1: Perfect negative correlation (as one variable increases, the other variable decreases proportionally).

As the example needs to be in a jupyter notebook, please check the IPYNB file present in the folder and search “Correlation”

**Machine learning**

**Question: 1**

Imagine you have a dataset where you have different Instagram features like u **sername , Caption , Hashtag , Followers , Time\_Since\_posted , and likes ,** now your task is to predict the number of likes and Time Since posted and the rest of the features are your input features. Now you have to build a model which can predict the number of likes and Time Since posted.

[Dataset](https://www.kaggle.com/datasets/rxsraghavagrawal/instagram-reach) This is the Dataset You can use this dataset for this question.

**ANS. Check the repository notebook file ->** [**soodsid/instagramquestion (github.com)**](https://github.com/soodsid/instagramquestion)

**Question: 2**

1. Train an SVM regressor on : [Bengaluru housing dataset](https://www.kaggle.com/datasets/amitabhajoy/bengaluru-house-price-data)

**Must include in details:**

- EDA

- Feature engineering

**ANS. Check the repository notebook file ->** [**soodsid/ml-question-2 (github.com)**](https://github.com/soodsid/ml-question-2)

**Question: 3**

Train and fine tune a decision tree using the wine dataset by following the following steps:-

1. Use load\_wine() to generate wine dataset

2. Split the dataset into train and test dataset

3. Use random search CV to hyperparameter tune the Decision Tree

4. Try to achieve an accuracy of at least 85%

Grow a random forest using the following steps:-

1. Continuing the previous question, create 10 subsets of the training dataset. You can use the ShuffleSplit class for it.

2. Train 1 decision tree on each subset, using the best hyperparameter values found in the previous question.

3. Evaluate all the trees on the test dataset. Are they performing better than the tree created in the previous question?

**ANS. Check the repository notebook file ->** [**soodsid/ml-question-no-3 (github.com)**](https://github.com/soodsid/ml-question-no-3)

**Deep Learning :**

**Question: 1**

(a) Explain how you can implement DL in a real-world application.

Ans.

Implementing Deep Learning (DL) in a real-world application involves several steps:

* Define the Problem: Clearly define the problem you want to solve and determine if DL is the right approach. DL is suitable for tasks such as image and speech recognition, natural language processing, and more.
* Collect and Preprocess Data: Gather a sufficient amount of labeled data for training and testing. Preprocess the data to ensure it is in a suitable format and is representative of the real-world scenarios.
* Choose a DL Framework: Select a deep learning framework such as TensorFlow, PyTorch, or Keras. These frameworks provide a set of tools and abstractions to simplify the implementation of neural networks.
* Design the Neural Network Architecture: Define the architecture of your neural network. This includes the number and type of layers, the activation functions, and the connections between neurons.
* Train the Model: Split your dataset into training and testing sets. Train the model on the training set using an optimization algorithm, adjusting the weights and biases of the network to minimize the error
* Validate and Tune: Evaluate the model on the validation set to ensure it generalizes well to new data. Fine-tune hyperparameters and architecture based on performance
* Deploy the Model: Once satisfied with the model's performance, deploy it to the real-world environment. This could involve integrating it into a web application, a mobile app, or an embedded system.
* Monitor and Update: Regularly monitor the model's performance in the real-world environment. If necessary, update the model with new data and retrain it to adapt to changing conditions.

(b) What is the use of Activation function in Artificial Neural Networks? What would be the problem if we don't use it in ANN networks.

Ans.

Activation functions introduce non-linearity in artificial neural networks (ANNs), enabling them to learn complex patterns and relationships in data. Without activation functions, ANNs would be limited to representing only linear functions, leading to poor performance and an inability to learn effectively. Additionally, activation functions help mitigate the vanishing gradient problem during training by allowing the network to capture more complex relationships and avoid saturation. Overall, activation functions are essential for the successful training and performance of ANNs.

* Introducing Non-linearity: Activation functions introduce non-linearity into the network, allowing it to learn and represent complex patterns and relationships in the data. Without non-linear activation functions, the entire network would simply be a linear combination of the input features, making it incapable of learning complex mappings between inputs and outputs.
* Enabling Representation of Complex Functions: Neural networks are universal function approximators, meaning they can approximate any continuous function given the right architecture and parameters. Non-linear activation functions are essential for enabling neural networks to represent and approximate complex functions, including those that are highly non-linear and non-continuous.
* Avoiding Gradient Vanishing/Exploding: During the process of backpropagation, which is used to update the network's weights based on the error between predicted and actual outputs, the gradient of the loss function with respect to the weights is computed. In deep neural networks, without proper activation functions, gradients can either vanish (become extremely small) or explode (become extremely large), making it challenging for the network to learn effectively. Non-linear activation functions help mitigate the vanishing gradient problem by allowing the network to capture more complex relationships and avoid saturation.

**Question: 2**

Train a Pure ANN with less than 10000 trainable parameters using the MNIST Dataset

**ANS. Check the repository notebook file ->** [**soodsid/DL-question-2 (github.com)**](https://github.com/soodsid/DL-question-2)

**Question: 3**

Perform Regression Task using ANN

**Note**: You are feel free to use any Regression ML dataset

**ANS. Check the repository notebook file ->** [**soodsid/DL\_question-3 (github.com)**](https://github.com/soodsid/DL_question-3)