EXAM PREPARATION 1

1. Gunakan Database World, Tampilkan seluruh bahasa yg digunakan di Timur tengah (Middle East)
    a. Select language as Bahasa, region as Region FROM Country LEFT JOIN CountryLanguage ON CountryLanguage.CountryCode = country.code Where country.region = 'Middle East';
    b. Select language as Bahasa, region as Region FROM Country LEFT JOIN CountryLanguage IN CountryLanguage.CountryCode = country.code Where country.region = 'Middle East';
    c. Select language as Bahasa, region as Region FROM Country LEFT JOIN CountryLanguage ON CountryLanguage.CountryCode = country.code Where country.continent = 'Middle East';
    d. Select language as Bahasa, region as Region FROM Country LEFT JOIN CountryLanguage ON CountryLanguage.CountryCode == country.code Where country.region = 'Middle East';
    e. Select language as Bahasa, region as Region FROM Country LEFT JOIN CountryLanguage ON CountryLanguage.CountryCode = country.CountryCode Where country.region = 'Middle East';
2. Query apa yg digunakan untuk menampilkan rata-rata GNP dari setiap Benua yg memiliki karakter E, diurutkan dari yg rata-rata GNP terkecil.
    a. Select Continent, avg(GNP) as Rerata FROM country group by 1 having continent like '%e' order by 2;
    b. Select Continent, avg(GNP) as Rerata FROM country group by 1 having continent like '%e%' order by 2;
    c. Select Continent, avg(GNP) as Rerata FROM country group by 1 having continent like 'e%' order by 2 desc;
    d. Select Continent, avg(GNP) as Rerata FROM country group by 1 having continent like '%e%' order by 2 desc;
    e. Select Continent, avg(GNP) as Rerata FROM country group by 1 where continent like '%e%' order by 1 desc;
3. Query yg digunakan untuk mencari Total jumlah Distrik di negara India adalah
    a. Select c.name, count(distinct ct.district) as Jumlah_Distrik from country c INNER JOIN city ct ON c.code = ct.countrycode GROUP BY c.name HAVING c.name = 'India';
    b. Select ct.name, count(distinct ct.district) as Jumlah_Distrik from country c INNER JOIN city ct ON c.code = ct.countrycode GROUP BY c.name HAVING c.name = 'India';
    c. Select c.name, count(distinct ct.district) as Jumlah_Distrik country c INNER JOIN city ct ON c.code = ct.countrycode GROUP BY c.name HAVING c.name = 'India';
    d. Select c.name, count(distinct ct.district) as Jumlah_Distrik from country c INNER JOIN city ct IN c.code = ct.countrycode GROUP BY c.name HAVING c.name = 'India';
    e. Select c.name, sum(distinct ct.district) as Jumlah_Distrik from country c INNER JOIN city ct ON c.code = ct.countrycode GROUP BY c.name where c.name = 'India';

4. Query yg digunakan untuk menampilkan 5 kota di benua asia dengan populasi terbanyak adalah
    a. select ct.name, ct.population as Population FROM city ct JOIN country c ON ct.countrycode = c.code Where c.Continent = 'ASIA' group by 2 desc Limit 5;
    b. select ct.name, ct.population as Population FROM city ct JOIN country c IN ct.countrycode = c.code Where c.Continent = 'ASIA' order by 2 desc Limit 5;
    c. select ct.name, ct.population as Population FROM city ct JOIN country c ON ct.countrycode = c.code Where c.Continent = 'ASIA' order by 2 Limit 5;
    d. select ct.name, ct.population as Population FROM city ct JOIN country c ON ct.countrycode = c.code Where c.Continent = 'ASIA' order by 2 desc Limit 5;
    e. select ct.name, ct.population as Population FROM city ct JOIN country c ON ct.countrycode = c.code Where c.Continent = 'ASIA' order by 1 desc Limit 5;
5. Kita memiliki array – arr sebagai berikut

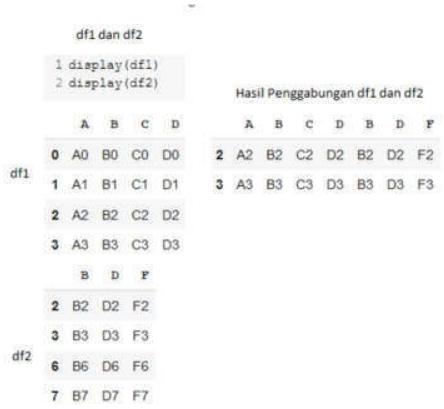
```
Out[4]: array([[ 1,  2,  3,  4,  5,  6,  7],
               [ 8,  9, 10, 11, 12, 13, 14],
               [15, 16, 17, 18, 19, 20, 21],
               [22, 23, 24, 25, 26, 27, 28],
               [29, 30, 31, 32, 33, 34, 35]])
```

Code yg digunakan jika ingin mengakses matriks berikut adalah

```
Out[6]: array([[11, 12, 13],
               [18, 19, 20],
               [25, 26, 27]])
```

    a. arr[np.array([1, 2, 3]), -4:-1]
    b. arr[np.array([1, 2, 3]), 3:6]
    c. arr[np.array([-4, -3, -2]), 3:6]
    d. arr[np.array([-4, -3, -2]), -4:-1]
    e. semua benar
6. Code apa yg digunakan untuk membuat angka random dari 0 sampai 1 dan tidak terdistribusi normal, dengan dimensi/shape 5 baris dan 6 kolom.
    a. np.random.rand(6,5)
    b. np.random.rand(5,6)
    c. np.random.randn(6,5)
    d. np.random.randn(5,6)
    e. np.random.randint(6,5)

7. Code yg digunakan untuk menggabungkan dataframe seperti di bawah adalah



   a. pd.concat([df1, df2], axis=1, join='inner')
   b. pd.concat([df1, df2], axis=0, join='outer')
   c. pd.concat([df1, df2], axis=1, join='left')
   d. pd.concat([df1, df2], axis=1, join='right')
   e. pd.concat([df1, df2], axis=0, join='inner')
8. Pada dataframe mpg, code yg digunakan untuk mencari jumlah cylinder yg unik adalah
   a. len(df['cylinders'].unique())
   b. len(pd.unique(df['cylinders']))
   c. df['cylinders'].nunique()
   d. len(df['cylinders'].value_counts())
   e. Semua benar
9. Import dataset mpg dari seaborn ke dalam variabel df, Code yg digunakan untuk mendapatkan median dari horse power, pada tahun model 76 adalah?
   a. df[df['model_year'] == 76].median()['horsepower']
   b. df[df['model_year'] == 76] ['horsepower'] .median()
   c. df['horsepower'][df['model_year'] == 76].median()
   d. A, B, C Salah
   e. A, B, C benar
10. Gunakan df dari soal diatas, Output dari code df.groupby('cylinders').median().sort_values('horsepower', ascending=False)['horsepower'].iloc[:3] adalah
   a. Median horse power untuk setiap jenis cylinders
   b. 3 jenis cylinders dan median horse power yg memiliki horsepower terendah.
   c. 3 jenis cylinders dan median horse power yg memiliki horsepower tertinggi.
   d. 3 jenis cylinders dan median horse power yg memiliki median horsepower terendah.
   e. 3 jenis cylinders dan median horse power yg memiliki median horsepower tertinggi.