



# Data Mining

تكليف مقرر تنقيب بيانات

القسم العملي

تكليف مقرر تنقيب البيانات

القسم العملي

مقدم من

محمد عبده ثابت محمد

كتكليف مقدم الى

قسم علوم الحاسوب وتقنية المعلومات المستوى

المستوى الثالث

اشراف

أ/ مالك المصنف

كتكليف لمقرر تنقيب البيانات

المحاضرة الرابعة

المجموعة B

DATA ITEGREATION

2024

# Data Integration



## أبرز المشكلات التي تم حلها في عملية تكامل البيانات

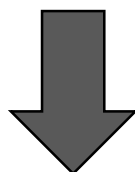
الرقم	المشكلة	الحل
1	-مشكلة اختلاف تنسيقات البيانات: البيانات تأتي من مصادر مختلفة (CSV, JSON, Excel) ولكل منها هيكل مختلف من الأعمدة والتنسيقات	-استخدم pd.read_csv و pd.read_excel و pd.read_json لتحميل البيانات من مصادر متعددة.
2	-مشكلة الأعمدة غير الضرورية: أعمدة غير مفيدة مثل "Unnamed"	-حذف الأعمدة غير المرغوب فيها
3	-مشكلة أسماء الأعمدة الغير متناسقة بين الملفات	-توحيد أسماء الأعمدة لضمان التناسق بين الملفات
4	●المشكلة: • بعض البيانات مكتوبة بصيغ مختلفة عبر الملفات. ○ مثل Diesel مقابل Petrol في ملف و 1, 2, 0 في ملف آخر. ○ بعض الأعمدة مثل Doors تحتوي على قيم بالعربية والإنجليزية.	-تحويل القيم النصية إلى أرقام -توحيد تمثيل الألوان(Paint_Type)

<ul style="list-style-type: none"> <li>• إنشاء فئة سعرية (Low, Medium, High) بناءً على Car_Price</li> </ul>	<p>- مشكلة القيم النصية التي تحتاج إلى تصنيف</p> <p>● المشكلة:</p> <ul style="list-style-type: none"> <li>• بعض البيانات مثل Price_Category تعتمد على قيم رقمية، لكنها غير مصنفة.</li> </ul>	5
<p>- حساب عمر السيارة تقديرًا على افتراض أن السيارة تقطع 5000 كم سنويًا</p>	<p>- مشكلة عدم وجود بيانات مثل عمر السيارة</p> <p>● المشكلة:</p> <ul style="list-style-type: none"> <li>• dataset3 لا يحتوي على عمر السيارة (Vehicle_Age)، لكنه يحتوي على عدد الكيلومترات (KM_Travelled).</li> </ul>	6

-لتحميل الكود الخاص بالمثال السابق اضغط على الرابط التالي:

<https://github.com/swdesign2024/Data-integreation-HW.git>

الكود



```
import pandas as pd
```

```
dataset1=pd.read_csv("DataSets/Toyta1.csv")
```

```
dataset1
```

	Unnamed: 0	Car_Price	Vehicle_Age	KM_Travelled	Fuel_Type	
HP \						
0	0	13500	23	46986	Diesel	90
1	1	13750	23	72937	Diesel	90
2	2	13950	24	41711	Diesel	90
3	3	14950	26	48000	Diesel	90
4	4	13750	30	38500	Diesel	90
..	...	...	...	...	...	...
473	473	11950	56	65000	Petrol	110
474	474	10450	48	64193	Petrol	110
475	475	8950	54	64000	Petrol	97
476	476	10250	54	63792	Petrol	110
477	477	9930	53	63635	Petrol	110

	Paint_Type	Transmission_Type	Engine_Size	Doors	Weight	
Age_Group						
0	Metallic	Manual	2000	1165.0	ثلاثة	Old
1	Metallic	Manual	2000	1165.0	ثلاثة	Old
2	Metallic	Manual	2000	1165.0	ثلاثة	Old
3	Non-Metallic	Manual	2000	1165.0	ثلاثة	Old
4	Non-Metallic	Manual	2000	1170.0	ثلاثة	Old
..	...	...	...	...	...	...
...						
473	Metallic	Manual	1600	1075.0	خمسة	Old
474	Metallic	Manual	1600	1040.0	ثلاثة	Old
475	Metallic	Manual	1400	1025.0	ثلاثة	Old

476	Metallic	Manual	1600	1075.0	خمسة
Old					
477	Metallic	Manual	1600	1035.0	أربعة
Old					

[478 rows x 12 columns]

```
dataset2=pd.read_json("DataSets/Toyta2.json")
dataset2
```

	Cost	Age_in_Years	Total_KM	FuelClass	HP	Body_Color	\
0	10500	54	63135	1	110	Main	
1	11950	54	63123	1	110	Main	
2	11500	55	63000	0	69	Main	
3	11500	55	63000	1	110	Alternative	
4	11450	54	62987	1	110	Alternative	
...	...	...	...	...	...	...	
473	8950	57	52548	1	110	Alternative	
474	8400	60	52487	1	110	Main	
475	9250	66	52383	1	86	Alternative	
476	8900	61	52112	1	110	Main	
477	8750	58	51712	1	110	Alternative	

	Transmission_Type	Engine_Size	Doors	Weight	Price_Category
Age_Group					
0	Manual	1600	three	1050	Medium
0					
1	Manual	1600	four	1035	Medium
0					
2	Manual	1900	five	1140	Medium
0					
3	Manual	1600	four	1035	Medium
0					
4	Manual	1600	five	1080	Medium
0					
...	...	...	...	...	...
...					
473	Manual	1600	three	1050	Low
0					
474	Manual	1600	four	1035	Low
0					
475	Manual	1300	three	1015	Low
0					
476	Manual	1600	four	1035	Low
0					
477	Manual	1600	three	1050	Low
0					

[478 rows x 12 columns]

```
dataset3=pd.read_excel("DataSets/Toyota3.xlsx")
dataset3
```

	Unnamed: 0.2	Unnamed: 0.1	Unnamed: 0	Sale_Price	Kilometers \
0	0	0	956	10950	51421
1	1	1	957	8950	51235
2	2	2	958	8950	51000
3	3	3	959	8895	50925
4	4	4	960	9390	50806
...	...	...	...	...	...
475	475	475	1431	7500	20544
476	476	476	1432	10845	11000
477	477	477	1433	8500	17016
478	478	478	1434	7250	11000
479	479	479	1435	6950	1

	Energy_Source	HP	Exterior_Finish	Transmission_Type	Engine_Size
Doors \					
0	1	110	Secondary	Auto	1600
5					
1	1	86	Primary	Manual	1300
4					
2	1	86	Primary	Manual	1300
3					
3	1	110	Primary	Manual	1600
5					
4	1	86	Secondary	Manual	1300
3					
...	...	...	...	...	...
...					
475	1	86	Primary	Manual	1300
3					
476	1	86	Secondary	Manual	1300
3					
477	1	86	Secondary	Manual	1300
3					
478	1	86	Primary	Manual	1300
3					
479	1	110	Secondary	Manual	1600
5					

	Weight	Price_Category	Random_Feature
0	1105	Medium	E
1	1000	Low	B
2	1015	Low	C
3	1070	Low	B
4	1480	Low	D
...	...	...	...
475	1025	Low	C
476	1015	Medium	B

477	1015	Low	B
478	1015	Low	D
479	1114	Low	D

[480 rows x 14 columns]

dataset1.isnull().sum()

Unnamed: 0	0
Car_Price	0
Vehicle_Age	0
KM_Travelled	0
Fuel_Type	0
HP	0
Paint_Type	0
Transmission_Type	0
Engine_Size	0
Doors	0
Weight	0
Age_Group	0

dtype: int64

dataset2.isnull().sum()

Cost	0
Age_in_Years	0
Total_KM	0
FuelClass	0
HP	0
Body_Color	0
Transmission_Type	0
Engine_Size	0
Doors	0
Weight	0
Price_Category	0
Age_Group	0

dtype: int64

dataset3.isnull().sum()

Unnamed: 0.2	0
Unnamed: 0.1	0
Unnamed: 0	0
Sale_Price	0
Kilometers	0
Energy_Source	0
HP	0
Exterior_Finish	0
Transmission_Type	0
Engine_Size	0
Doors	0

```
Weight          0
Price_Category  0
Random_Feature  0
dtype: int64
```

```
dataset1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 478 entries, 0 to 477
Data columns (total 12 columns):
#   Column              Non-Null Count  Dtype
---  -
0   Unnamed: 0          478 non-null   int64
1   Car_Price           478 non-null   int64
2   Vehicle_Age         478 non-null   int64
3   KM_Travelled        478 non-null   int64
4   Fuel_Type           478 non-null   object
5   HP                  478 non-null   int64
6   Paint_Type          478 non-null   object
7   Transmission_Type   478 non-null   object
8   Engine_Size         478 non-null   int64
9   Doors               478 non-null   object
10  Weight              478 non-null   float64
11  Age_Group           478 non-null   object
dtypes: float64(1), int64(6), object(5)
memory usage: 44.9+ KB
```

```
dataset2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 478 entries, 0 to 477
Data columns (total 12 columns):
#   Column              Non-Null Count  Dtype
---  -
0   Cost                478 non-null   int64
1   Age_in_Years        478 non-null   int64
2   Total_KM            478 non-null   int64
3   FuelClass           478 non-null   int64
4   HP                  478 non-null   int64
5   Body_Color          478 non-null   object
6   Transmission_Type   478 non-null   object
7   Engine_Size         478 non-null   int64
8   Doors               478 non-null   object
9   Weight              478 non-null   int64
10  Price_Category      478 non-null   object
11  Age_Group           478 non-null   int64
dtypes: int64(8), object(4)
memory usage: 44.9+ KB
```

```
dataset3.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 480 entries, 0 to 479
Data columns (total 14 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Unnamed: 0.2          480 non-null   int64
1   Unnamed: 0.1          480 non-null   int64
2   Unnamed: 0            480 non-null   int64
3   Sale_Price            480 non-null   int64
4   Kilometers            480 non-null   int64
5   Energy_Source         480 non-null   int64
6   HP                    480 non-null   int64
7   Exterior_Finish       480 non-null   object
8   Transmission_Type     480 non-null   object
9   Engine_Size           480 non-null   int64
10  Doors                 480 non-null   int64
11  Weight                480 non-null   int64
12  Price_Category        480 non-null   object
13  Random_Feature        480 non-null   object
dtypes: int64(10), object(4)
memory usage: 52.6+ KB
```

Drop The Columns That Unnamed in Dataset

```
dataset1.drop(columns=['Unnamed: 0'],inplace=True)
dataset1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 478 entries, 0 to 477
Data columns (total 11 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Car_Price             478 non-null   int64
1   Vehicle_Age           478 non-null   int64
2   KM_Travelled          478 non-null   int64
3   Fuel_Type             478 non-null   object
4   HP                    478 non-null   int64
5   Paint_Type            478 non-null   object
6   Transmission_Type     478 non-null   object
7   Engine_Size           478 non-null   int64
8   Doors                 478 non-null   object
9   Weight                478 non-null   float64
10  Age_Group             478 non-null   object
dtypes: float64(1), int64(5), object(5)
memory usage: 41.2+ KB
```

```
dataset3.drop(columns=['Unnamed: 0.2'],inplace=True)
dataset3.drop(columns=['Unnamed: 0.1'],inplace=True)
```

```
dataset3.drop(columns=['Unnamed: 0'],inplace=True)
dataset3.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 480 entries, 0 to 479
Data columns (total 11 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Sale_Price            480 non-null    int64
1   Kilometers            480 non-null    int64
2   Energy_Source         480 non-null    int64
3   HP                    480 non-null    int64
4   Exterior_Finish       480 non-null    object
5   Transmission_Type     480 non-null    object
6   Engine_Size           480 non-null    int64
7   Doors                 480 non-null    int64
8   Weight                480 non-null    int64
9   Price_Category        480 non-null    object
10  Random_Feature        480 non-null    object
dtypes: int64(7), object(4)
memory usage: 41.4+ KB
```

```
print(dataset1.shape)
print(dataset2.shape)
print(dataset3.shape)
```

```
(478, 11)
(478, 12)
(480, 11)
```

```
dataset1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 478 entries, 0 to 477
Data columns (total 11 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Car_Price             478 non-null    int64
1   Vehicle_Age           478 non-null    int64
2   KM_Travelled          478 non-null    int64
3   Fuel_Type             478 non-null    object
4   HP                    478 non-null    int64
5   Paint_Type            478 non-null    object
6   Transmission_Type     478 non-null    object
7   Engine_Size           478 non-null    int64
8   Doors                 478 non-null    object
9   Weight                478 non-null    float64
10  Age_Group             478 non-null    object
dtypes: float64(1), int64(5), object(5)
memory usage: 41.2+ KB
```

```
dataset2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 478 entries, 0 to 477
```

```
Data columns (total 12 columns):
```

#	Column	Non-Null Count	Dtype
0	Cost	478 non-null	int64
1	Age_in_Years	478 non-null	int64
2	Total_KM	478 non-null	int64
3	FuelClass	478 non-null	int64
4	HP	478 non-null	int64
5	Body_Color	478 non-null	object
6	Transmission_Type	478 non-null	object
7	Engine_Size	478 non-null	int64
8	Doors	478 non-null	object
9	Weight	478 non-null	int64
10	Price_Category	478 non-null	object
11	Age_Group	478 non-null	int64

```
dtypes: int64(8), object(4)
```

```
memory usage: 44.9+ KB
```

```
dataset2.rename(columns={  
    'Cost': 'Car_Price',  
    'Age_in_Years': 'Vehicle_Age',  
    'Total_KM': 'KM_Travelled',  
    'FuelClass': 'Fuel_Type',  
    'Body_Color': 'Paint_Type'  
}, inplace=True)
```

```
dataset3.columns
```

```
Index(['Sale_Price', 'Kilometers', 'Energy_Source', 'HP',  
      'Exterior_Finish',  
      'Transmission_Type', 'Engine_Size', 'Doors', 'Weight',  
      'Price_Category',  
      'Random_Feature'],  
      dtype='object')
```

```
dataset3.rename(columns={  
    'Sale_Price': 'Car_Price',  
    'Kilometers': 'KM_Travelled',  
    'Energy_Source': 'Fuel_Type',  
    'Exterior_Finish': 'Paint_Type'  
}, inplace=True)
```

```
dataset1.columns
```

```
Index(['Car_Price', 'Vehicle_Age', 'KM_Travelled', 'Fuel_Type', 'HP',  
      'Paint_Type', 'Transmission_Type', 'Engine_Size', 'Doors',  
      'Weight',
```

```
'Age_Group'],  
dtype='object')
```

```
dataset2.columns
```

```
Index(['Car_Price', 'Vehicle_Age', 'KM_Travelled', 'Fuel_Type', 'HP',  
      'Paint_Type', 'Transmission_Type', 'Engine_Size', 'Doors',  
      'Weight',  
      'Price_Category', 'Age_Group'],  
      dtype='object')
```

```
dataset3.columns
```

```
Index(['Car_Price', 'KM_Travelled', 'Fuel_Type', 'HP', 'Paint_Type',  
      'Transmission_Type', 'Engine_Size', 'Doors', 'Weight',  
      'Price_Category',  
      'Random_Feature'],  
      dtype='object')
```

```
dataset1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 478 entries, 0 to 477
```

```
Data columns (total 11 columns):
```

#	Column	Non-Null Count	Dtype
0	Car_Price	478 non-null	int64
1	Vehicle_Age	478 non-null	int64
2	KM_Travelled	478 non-null	int64
3	Fuel_Type	478 non-null	object
4	HP	478 non-null	int64
5	Paint_Type	478 non-null	object
6	Transmission_Type	478 non-null	object
7	Engine_Size	478 non-null	int64
8	Doors	478 non-null	object
9	Weight	478 non-null	float64
10	Age_Group	478 non-null	object

```
dtypes: float64(1), int64(5), object(5)
```

```
memory usage: 41.2+ KB
```

```
dataset2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 478 entries, 0 to 477
```

```
Data columns (total 12 columns):
```

#	Column	Non-Null Count	Dtype
0	Car_Price	478 non-null	int64
1	Vehicle_Age	478 non-null	int64
2	KM_Travelled	478 non-null	int64
3	Fuel_Type	478 non-null	int64

```

4   HP                478 non-null    int64
5   Paint_Type        478 non-null    object
6   Transmission_Type 478 non-null    object
7   Engine_Size        478 non-null    int64
8   Doors              478 non-null    object
9   Weight             478 non-null    int64
10  Price_Category     478 non-null    object
11  Age_Group          478 non-null    int64
dtypes: int64(8), object(4)
memory usage: 44.9+ KB

dataset1.Fuel_Type.unique()

array(['Diesel', 'Petrol', 'CNG'], dtype=object)

dataset2.Fuel_Type.unique()

array([1, 0, 2], dtype=int64)

for i in range(len(dataset1['Fuel_Type'])):
    if dataset1.loc[i, 'Fuel_Type']=='Diesel':
        dataset1.loc[i, 'Fuel_Type']=0
    elif dataset1.loc[i, 'Fuel_Type']=='Petrol':
        dataset1.loc[i, 'Fuel_Type']=1
    else:
        dataset1.loc[i, 'Fuel_Type']=2

dataset1['Fuel_Type']=dataset1['Fuel_Type'].astype('int64')

dataset1.Fuel_Type.unique()

array([0, 1, 2], dtype=int64)

dataset1['Weight']=dataset1['Weight'].astype('int64')

dataset1.Doors.unique()

array(['أربعة', 'خمسة', 'ثلاثة'], dtype=object)

for i in range(len(dataset1['Doors'])):
    if dataset1.loc[i, 'Doors']=='ثلاثة':
        dataset1.loc[i, 'Doors']=3
    elif dataset1.loc[i, 'Doors']=='أربعة':
        dataset1.loc[i, 'Doors']=4
    else:
        dataset1.loc[i, 'Doors']=5

dataset1.Doors.unique()

array([3, 5, 4], dtype=object)

dataset1['Doors']=dataset1['Doors'].astype('int64')

```

```

dataset2.Doors.unique()
array(['three', 'four', 'five', 'two'], dtype=object)

for i in range(len(dataset2['Doors'])):
    if dataset2.loc[i, 'Doors']=='two':
        dataset2.loc[i, 'Doors']=2
    elif dataset2.loc[i, 'Doors']=='three':
        dataset2.loc[i, 'Doors']=3
    elif dataset2.loc[i, 'Doors']=='four':
        dataset2.loc[i, 'Doors']=4
    else:
        dataset2.loc[i, 'Doors']=5

dataset2['Doors']=dataset2['Doors'].astype('int64')
dataset2.Doors.unique()
array([3, 4, 5, 2], dtype=int64)

dataset1.Age_Group.unique()
array(['Old', 'New', 'Moderate'], dtype=object)

dataset2.Age_Group.unique()
array([0], dtype=int64)

for i in range(len(dataset1['Age_Group'])):
    if dataset1.loc[i, 'Age_Group']=='Old':
        dataset1.loc[i, 'Age_Group']=0
    elif dataset1.loc[i, 'Age_Group']=='New':
        dataset1.loc[i, 'Age_Group']=2
    else:
        dataset1.loc[i, 'Age_Group']=1

dataset1['Age_Group']=dataset1['Age_Group'].astype('int64')
dataset1.Age_Group.unique()
array([0, 2, 1], dtype=int64)

dataset1.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 478 entries, 0 to 477
Data columns (total 11 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Car_Price              478 non-null    int64
1   Vehicle_Age            478 non-null    int64
2   KM_Travelled           478 non-null    int64
3   Fuel_Type              478 non-null    int64

```

```
4    HP          478 non-null    int64
5    Paint_Type  478 non-null    object
6    Transmission_Type 478 non-null    object
7    Engine_Size 478 non-null    int64
8    Doors       478 non-null    int64
9    Weight      478 non-null    int64
10   Age_Group   478 non-null    int64
dtypes: int64(9), object(2)
memory usage: 41.2+ KB
```

```
dataset2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 478 entries, 0 to 477
Data columns (total 12 columns):
#   Column              Non-Null Count  Dtype
---  -
0   Car_Price           478 non-null    int64
1   Vehicle_Age         478 non-null    int64
2   KM_Travelled        478 non-null    int64
3   Fuel_Type           478 non-null    int64
4   HP                  478 non-null    int64
5   Paint_Type          478 non-null    object
6   Transmission_Type    478 non-null    object
7   Engine_Size         478 non-null    int64
8   Doors               478 non-null    int64
9   Weight              478 non-null    int64
10  Price_Category       478 non-null    object
11  Age_Group            478 non-null    int64
dtypes: int64(9), object(3)
memory usage: 44.9+ KB
```

```
dataset3.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 480 entries, 0 to 479
Data columns (total 11 columns):
#   Column              Non-Null Count  Dtype
---  -
0   Car_Price           480 non-null    int64
1   KM_Travelled        480 non-null    int64
2   Fuel_Type           480 non-null    int64
3   HP                  480 non-null    int64
4   Paint_Type          480 non-null    object
5   Transmission_Type    480 non-null    object
6   Engine_Size         480 non-null    int64
7   Doors               480 non-null    int64
8   Weight              480 non-null    int64
9   Price_Category       480 non-null    object
10  Random_Feature       480 non-null    object
```

```
dtypes: int64(7), object(4)
```

```
memory usage: 41.4+ KB
```

```
dataset3.drop(columns=['Random_Feature'],inplace=True)
```

```
dataset1.head(5)
```

	Car_Price	Vehicle_Age	KM_Travelled	Fuel_Type	HP	Paint_Type \
0	13500	23	46986	0	90	Metallic
1	13750	23	72937	0	90	Metallic
2	13950	24	41711	0	90	Metallic
3	14950	26	48000	0	90	Non-Metallic
4	13750	30	38500	0	90	Non-Metallic

	Transmission_Type	Engine_Size	Doors	Weight	Age_Group
0	Manual	2000	3	1165	0
1	Manual	2000	3	1165	0
2	Manual	2000	3	1165	0
3	Manual	2000	3	1165	0
4	Manual	2000	3	1170	0

```
dataset2.head(5)
```

	Car_Price	Vehicle_Age	KM_Travelled	Fuel_Type	HP	Paint_Type \
0	10500	54	63135	1	110	Main
1	11950	54	63123	1	110	Main
2	11500	55	63000	0	69	Main
3	11500	55	63000	1	110	Alternative
4	11450	54	62987	1	110	Alternative

	Transmission_Type	Engine_Size	Doors	Weight	Price_Category
0	Manual	1600	3	1050	Medium
0					
1	Manual	1600	4	1035	Medium
0					
2	Manual	1900	5	1140	Medium
0					
3	Manual	1600	4	1035	Medium

0					
4	Manual	1600	5	1080	Medium
0					

```
dataset3.head(5)
```

	Car_Price	KM_Travelled	Fuel_Type	HP	Paint_Type
Transmission_Type \					
0	10950	51421	1	110	Secondary
Auto					
1	8950	51235	1	86	Primary
Manual					
2	8950	51000	1	86	Primary
Manual					
3	8895	50925	1	110	Primary
Manual					
4	9390	50806	1	86	Secondary
Manual					

	Engine_Size	Doors	Weight	Price_Category
0	1600	5	1105	Medium
1	1300	4	1000	Low
2	1300	3	1015	Low
3	1600	5	1070	Low
4	1300	3	1480	Low

```
dataset1.Paint_Type.unique()
```

```
array(['Metallic', 'Non-Metallic'], dtype=object)
```

```
dataset2.Paint_Type.unique()
```

```
array(['Main', 'Alternative'], dtype=object)
```

```
dataset3.Paint_Type.unique()
```

```
array(['Secondary', 'Primary'], dtype=object)
```

```
for i in range(len(dataset2['Paint_Type'])):
    if dataset2.loc[i, 'Paint_Type']=='Main':
        dataset2.loc[i, 'Paint_Type']='Metallic'
    else:
        dataset2.loc[i, 'Paint_Type']='Non-Metallic'

for i in range(len(dataset3['Paint_Type'])):
    if dataset3.loc[i, 'Paint_Type']=='Primary':
        dataset3.loc[i, 'Paint_Type']='Metallic'
    else:
        dataset3.loc[i, 'Paint_Type']='Non-Metallic'
```

```
dataset3.Paint_Type.unique()
```

```

array(['Non-Metallic', 'Metallic'], dtype=object)

dataset3.Price_Category.unique()

array(['Medium', 'Low'], dtype=object)

def generate_category_price(price):
    if price < 5000:
        return 'Very Low'
    elif 5000 <= price <= 10000:
        return 'Low'
    elif 10001 <= price <= 19000:
        return 'Medium'
    else:
        return 'High'

# تطبيق الدالة على عمود البيانات
dataset1['Price_Category'] =
dataset1['Car_Price'].apply(generate_category_price)

dataset1.Price_Category.unique()

array(['Medium', 'High', 'Low', 'Very Low'], dtype=object)

import numpy as np

# نفترض أن السيارة تقطع حوالي 15,000 كم في السنة
km_per_year = 5000

# تقدير عمر السيارة باستخدام KM_Travelled
dataset3['Vehicle_Age'] = (dataset3['KM_Travelled'] /
km_per_year).apply(np.floor).astype(int)

# تقييد عمر السيارة بحيث لا يكون أقل من 2 ولا يزيد عن 50
dataset3['Vehicle_Age'] = dataset3['Vehicle_Age'].clip(lower=2,
upper=50)

# بناءً على العمر المقدر Age_Group إنشاء فئات
def categorize_age(age):
    if age < 2:
        return 2 # سيارة جديدة
    elif 2 <= age <= 5:
        return 1 # سيارة متوسطة العمر
    else:
        return 0 # سيارة قديمة

dataset3['Age_Group'] = dataset3['Vehicle_Age'].apply(categorize_age)

dataset3['Vehicle_Age']=dataset3['Vehicle_Age'].astype('int64')

```

```
dataset1.Age_Group.unique()  
array([0, 2, 1], dtype=int64)
```

```
dataset3.Age_Group.unique()  
array([0, 1], dtype=int64)
```

```
dataset1.head(5)
```

	Car_Price	Vehicle_Age	KM_Travelled	Fuel_Type	HP	Paint_Type \
0	13500	23	46986	0	90	Metallic
1	13750	23	72937	0	90	Metallic
2	13950	24	41711	0	90	Metallic
3	14950	26	48000	0	90	Non-Metallic
4	13750	30	38500	0	90	Non-Metallic

	Transmission_Type	Engine_Size	Doors	Weight	Age_Group	Price_Category
0	Manual	2000	3	1165	0	Medium
1	Manual	2000	3	1165	0	Medium
2	Manual	2000	3	1165	0	Medium
3	Manual	2000	3	1165	0	Medium
4	Manual	2000	3	1170	0	Medium

```
dataset2.head(5)
```

	Car_Price	Vehicle_Age	KM_Travelled	Fuel_Type	HP	Paint_Type	Transmission_Type	Engine_Size	Doors	Weight	Price_Category
0	10500	54	63135	1	110	Metallic					
1	11950	54	63123	1	110	Metallic					
2	11500	55	63000	0	69	Metallic					
3	11500	55	63000	1	110	Non-Metallic					
4	11450	54	62987	1	110	Non-Metallic					

Age_Group					
0	Manual	1600	3	1050	Medium
0					
1	Manual	1600	4	1035	Medium
0					
2	Manual	1900	5	1140	Medium
0					
3	Manual	1600	4	1035	Medium
0					
4	Manual	1600	5	1080	Medium
0					

dataset3.head(5)

	Car_Price	KM_Travelled	Fuel_Type	HP	Paint_Type
Transmission_Type \					
0	10950	51421	1	110	Non-Metallic
Auto					
1	8950	51235	1	86	Metallic
Manual					
2	8950	51000	1	86	Metallic
Manual					
3	8895	50925	1	110	Metallic
Manual					
4	9390	50806	1	86	Non-Metallic
Manual					

	Engine_Size	Doors	Weight	Price_Category	Vehicle_Age	Age_Group
0	1600	5	1105	Medium	10	0
1	1300	4	1000	Low	10	0
2	1300	3	1015	Low	10	0
3	1600	5	1070	Low	10	0
4	1300	3	1480	Low	10	0

dataset1.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 478 entries, 0 to 477
Data columns (total 12 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Car_Price              478 non-null    int64
1   Vehicle_Age            478 non-null    int64
2   KM_Travelled           478 non-null    int64
3   Fuel_Type              478 non-null    int64
4   HP                     478 non-null    int64
5   Paint_Type             478 non-null    object
6   Transmission_Type      478 non-null    object
7   Engine_Size            478 non-null    int64
8   Doors                  478 non-null    int64
```

```
9   Weight          478 non-null    int64
10  Age_Group       478 non-null    int64
11  Price_Category  478 non-null    object
```

```
dtypes: int64(9), object(3)
```

```
memory usage: 44.9+ KB
```

```
dataset2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 478 entries, 0 to 477
```

```
Data columns (total 12 columns):
```

#	Column	Non-Null Count	Dtype
0	Car_Price	478 non-null	int64
1	Vehicle_Age	478 non-null	int64
2	KM_Travelled	478 non-null	int64
3	Fuel_Type	478 non-null	int64
4	HP	478 non-null	int64
5	Paint_Type	478 non-null	object
6	Transmission_Type	478 non-null	object
7	Engine_Size	478 non-null	int64
8	Doors	478 non-null	int64
9	Weight	478 non-null	int64
10	Price_Category	478 non-null	object
11	Age_Group	478 non-null	int64

```
dtypes: int64(9), object(3)
```

```
memory usage: 44.9+ KB
```

```
dataset3.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 480 entries, 0 to 479
```

```
Data columns (total 12 columns):
```

#	Column	Non-Null Count	Dtype
0	Car_Price	480 non-null	int64
1	KM_Travelled	480 non-null	int64
2	Fuel_Type	480 non-null	int64
3	HP	480 non-null	int64
4	Paint_Type	480 non-null	object
5	Transmission_Type	480 non-null	object
6	Engine_Size	480 non-null	int64
7	Doors	480 non-null	int64
8	Weight	480 non-null	int64
9	Price_Category	480 non-null	object
10	Vehicle_Age	480 non-null	int64
11	Age_Group	480 non-null	int64

```
dtypes: int64(9), object(3)
```

```
memory usage: 45.1+ KB
```

```

dataset1.shape
(478, 12)
dataset2.shape
(478, 12)
dataset3.shape
(480, 12)

dataset3 = dataset3[dataset1.columns] # إعادة ترتيب الأعمدة لمطابقة
dataset1
newDataSet = pd.concat([dataset1, dataset2, dataset3],
ignore_index=True)

newDataSet.drop_duplicates(inplace=True)

newDataSet.shape
(1436, 12)

newDataSet.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1436 entries, 0 to 1435
Data columns (total 12 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Car_Price              1436 non-null  int64
1   Vehicle_Age            1436 non-null  int64
2   KM_Travelled           1436 non-null  int64
3   Fuel_Type              1436 non-null  int64
4   HP                     1436 non-null  int64
5   Paint_Type             1436 non-null  object
6   Transmission_Type      1436 non-null  object
7   Engine_Size            1436 non-null  int64
8   Doors                  1436 non-null  int64
9   Weight                 1436 non-null  int64
10  Age_Group              1436 non-null  int64
11  Price_Category         1436 non-null  object
dtypes: int64(9), object(3)
memory usage: 134.8+ KB

newDataSet.columns

Index(['Car_Price', 'Vehicle_Age', 'KM_Travelled', 'Fuel_Type', 'HP',
      'Paint_Type', 'Transmission_Type', 'Engine_Size', 'Doors',
      'Weight',
      'Age_Group', 'Price_Category'],
      dtype='object')

```

```
newDataSet.isnull().sum()
```

```
Car_Price      0
Vehicle_Age    0
KM_Travelled   0
Fuel_Type      0
HP             0
Paint_Type     0
Transmission_Type 0
Engine_Size    0
Doors          0
Weight         0
Age_Group      0
Price_Category 0
dtype: int64
```

```
newDataSet
```

	Car_Price	Vehicle_Age	KM_Travelled	Fuel_Type	HP	
Paint_Type \						
0	13500	23	46986	0	90	
Metallic						
1	13750	23	72937	0	90	
Metallic						
2	13950	24	41711	0	90	
Metallic						
3	14950	26	48000	0	90	Non-
Metallic						
4	13750	30	38500	0	90	Non-
Metallic						
...	...	...	...	...	...	.
..						
1431	7500	4	20544	1	86	
Metallic						
1432	10845	2	11000	1	86	Non-
Metallic						
1433	8500	3	17016	1	86	Non-
Metallic						
1434	7250	2	11000	1	86	
Metallic						
1435	6950	2	1	1	110	Non-
Metallic						

	Transmission_Type	Engine_Size	Doors	Weight	Age_Group
Price_Category					
0	Manual	2000	3	1165	0
Medium					
1	Manual	2000	3	1165	0
Medium					
2	Manual	2000	3	1165	0

Medium					
3	Manual	2000	3	1165	0
Medium					
4	Manual	2000	3	1170	0
Medium					
...	...	...	...	...	...
...					
1431	Manual	1300	3	1025	1
Low					
1432	Manual	1300	3	1015	1
Medium					
1433	Manual	1300	3	1015	1
Low					
1434	Manual	1300	3	1015	1
Low					
1435	Manual	1600	5	1114	1
Low					

[1436 rows x 12 columns]

`newDataSet.to_csv('IntgreteD_Datasets.csv')`