Date	7 November 2022
Team ID	PNT2022TMIDO1544
Project Name	PROJECT-CAR RESALES VALUE PREDICTION
Maximum Marks	2 Marks

## **Collect dataset:**

Machine Learning has become a tool used in almost every task that requires estimation. So we need to build a model to estimate the price of used cars. The model should take car-related parameters and output a selling price. On sprint-1 the selling price of a used car depends on certain features datasets are collected from different open sources like kaggle.com, data.gov, UCI machine learning repository, the dataset which contains a set of features through which the resale price of the car can be identified is to be collected as

- seller
- offerType
- price
- vehicleType
- yearOfRegistration
- gearbox
- powerPS
- model
- kilometer
- monthOfRegistration
- fuelType
- brand
- notRepairedDamage

ML is a data hunger technology, it depends heavily on data, without data, it is impossible. It is the most crucial aspect that makes algorithm training possible. Collects Data, Import necessary packages, Pre-process images, and passes on to Network Model and Saves Model Weights. The libraries can be imported,

[] import pandes of pd import namey as op import namey as op import namey as op import name in the import name in the import name from sakern, preprocessing import LabelIncoder import pickle import pickle import name impor																			
[ ] df = pd.read_csv(" df.head()	/content/drive/MyDrive/Colab No	stebooks/a	autos.csv")																
	name	seller	offerType	price	abtest	vehicleType	vearOfRegistration	gearbox	powerPS	model	kilometer	monthOfRegistration	fuelType	brand	notRepairedDamage	dateCreated	nrOfPictures	postalCode	lastSeen
dateCrawled 0 24-03-2016 11:52			offerType Angebot	price 480	abtest test	vehicleType NaN	yearOfRegistration	-			kilometer 150000.00	monthOfRegistration			notRepairedDamage NaN	dateCreated 24-03-2016 00:00	nrOfPictures		1astSeen 07-04-2016 03:16
dateCrawled	Golf_3_1.6	privat	Angebot	480			-	manuell	0	golf		-		volkswagen	NaN			70435	
dateCrawled 0 24-03-2016 11:52 1 24-03-2016 10:58	Golf_3_1.6	privat privat	Angebot	480 18300	test	NaN	1993 2011	manuell	190	golf NaN	150000.00	-	benzin diesel	volkswagen audi	NaN ja	24-03-2016 00:00	0.00	70435 66954	07-04-2016 03:16
dateCrawled 0 24-03-2016 11:52 1 24-03-2016 10:58	Golf_3_1.6 A5_Sportback_2.7_Tdi Jeep_Grand_Cherokee_"Overland"	privat privat privat	Angebot Angebot	480 18300 9800	test test	NaN coupe	1993 2011	manuell manuell automatik	0 190 163	golf NaN grand	150000.00 125000.00	5	benzin diesel diesel	volkswagen audi jeep	NaN ja NaN	24-03-2016 00:00 24-03-2016 00:00	0.00	70435 66954 90480	07-04-2016 03:16 07-04-2016 01:46

## **Pre-Process The Data:**

Pre-processing the dataset that includes:

- Handling the null values.
- Handling the categorical values if any.
- Normalize the data if required.
- Identify the dependent and independent variables.

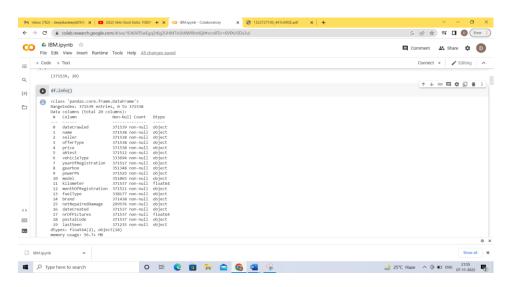
Data cleaning and wrangling methods are applied on the *used cars* data file. Before making data cleaning, some explorations and data visualizations were applied on data set. This gave some idea and guide about how to deal with missing values and extreme values. After data cleaning, data exploration was applied again in order to understand cleaned version of the data.

df = pd.read\_csv("/content/drive/MyDrive/Colab Notebooks/autos.csv")
df.head()



print(df.shape)
(371539, 20)

df.info()



```
df['powerPS'].unique()
                 32) - deepikadeepik09 🛭 🗴 | 🔼 (262) Vetri Kodi Kattu 1080 P 🔸 🗴 🕜 IBM.ipynb - Colaboratory 💢 😵 1523727100_441IJARSE.pdf 💢 🔭
  ← → C 🕯 colab.research.google.com/drive/1EtKAFl5wEgq2tKg2UHM7sUkNWIXml6jk#scrollTo=xm9jX4nhtMjU
                                                                                                                                                                                                                                       G 🖻 🖈 🗊 🔲 📵 Error 🗓
  CO & IBM.ipynb 🜣
                                                                                                                                                                                                                                         Comment A Share
            File Edit View Insert Runtime Tools Help All changes saved
                                                                                                                                                                                                                                                  + Code + Text
          odf['powerPS'].unique()
                  df['powerPS'].unique()

array(['e', '19e', '163', '75', '6e', '102', '10e', '5e', '125', '101', '105', '140', '115', '131', '6e', '126', '160', '231', '9e', '118', '193', '9e', '113', '218', '122', '129', '7e', '36e', '95', '61', '177', '8e', '170', '55', '143', '64', '28e', '232', '150', '156', '82', '264', '155', '54', '183', '87', '180', '86', '84', '303', '224', '235', '20e', '178', '265', '77', '11e', '1444', '12e', '161', '184', '126', '88', '144', '305', '197', '179', '25e', '45', '313', '41', '165', '98', '136', '114', '211', '56', '32e', '201', '213', '58', '107', '83', '174', '10e', '22e', '68', '66', '29e', '74', '52', '510', '147', '65', '31e', '71', '97', '239', '295', '293', '53', '530e', '103', '45', '258', '292', '320', '63', '81', '148', '354', '44', '145', '23e', '286', '26e', '457', '104', '40e', '188', '333', '186', '117', '141', '56', '32e', '272', '92', '51', '135', '53', '435', '226', '43', '146', '67', '106', '166', '276', '344', '349', '72', '249', '237', '311', '344', '128', '133', '124', '138', '329', '237', '301', 'andere', '334', '128', '133', '124', '33', '219', '221', '48', '139', '179', '40', '40', '40', '79', '36', '23', '301', '301', '134', '128', '391', '128', '46', '108', '129', '23', '37', '301', '344', '128', '129', '24', '158', '229', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '329', '32
                                                                                                                                                                                                                                                  ↑ ↓ © 目 ‡ 🖟 📋 :
 Q
 {x}
 >_
                                                                        O H C 🗓 🙀 😭 😘
                                                                                                                                                                                           25°C Haze ∧ 📴 🖭 ENG 21:56 📆
Type here to search
df=df[df['powerPS'].str.isnumeric().fillna(False)]
print(df.seller.value counts())
df[df.seller != 'gewerblich']
print(df.offerType.value counts())
df[df.offerType != 'Gesuch']
M Inbox (782) - deepikadeepik09⊚ x □ (262) Muthu Movie Songs □ 4 x CO IBM.ipynb - Colaboratory x ⑤ 1523727100_441IJARSE.pdf x □ +
  G Q 🖻 🖈 🗐 📵 (Error 🚦
€ IBM Jpynib th
File Est View Insert Runtime Tools Help <u>All chances</u>
      • print(df.seller.value_counts())
                                                                                                                                                                                                                                                                          * * * * * * * :
      dfidf[(df.powerPS > 50) & (df.powerPS < 900)]]
print(df.shape)</pre>
    [ ] df-df[(df.yearOfRegistration > 2950) & (df.yearOfRegistration < 2017)]
print(df.shape)
Type here to search
                                                                         O H C 🗓 ⊨ 😭 😘
                                                                                                                                                                                                                              df['powerPS'] = df['powerPS'].astype(int)
df=df[(df.powerPS > 50) & (df.powerPS < 900)]</pre>
print(df.shape)
df=df[df['yearOfRegistration'].str.isnumeric().fillna(False)]
```

```
df=df[(df.yearOfRegistration > 1950) & (df.yearOfRegistration < 2017)]</pre>
print(df.shape)
df.drop(['name', 'abtest', 'dateCrawled', 'nrOfPictures', 'lastSeen', '
postalCode', 'dateCreated'], axis='columns', inplace=True)
df.info()
  <class 'pandas.core.frame.DataFrame'>
  Int64Index: 308923 entries, 1 to 371538
  Data columns (total 13 columns):
                                                Dtype
       Column
                              Non-Null Count
   #
       -----
       seller
                              308923 non-null
                                                obiect
   0
     offerType
                                                object
   1
                              308923 non-null
      price
                              308923 non-null
                                                object
   2
   3
      vehicleType
                              297510 non-null
                                                object
      yearOfRegistration
                              308923 non-null
                                                int64
                              303629 non-null
                                                object
   5
       gearbox
                              308923 non-null
                                                int64
   6
       powerPS
       model
   7
                              297134 non-null
                                                object
      kilometer
                              308923 non-null float64
       monthOfRegistration 308923 non-null
                                                object
   10 fuelType
                              293046 non-null
                                                object
   11 brand
                              308923 non-null
                                                object
   12 notRepairedDamage
                              265507 non-null
                                                object
  dtypes: float64(1), int64(2), object(10)
  memory usage: 33.0+ MB
new df=df.copy()
new df = new df.drop duplicates(['price', 'vehicleType', 'yearOfRegistr
'gearbox', 'powerPS', 'model', 'kilometer', 'monthOfRegistration', 'fue
lType',
'notRepairedDamage'])
new df.gearbox.replace(('manuell', 'automatik'), ('manual', 'automatic'
), inplace=True)
new df.fuelType.replace(('benzin', 'andere', 'elektro'), ('petrol', 'ot
hers', 'electric'), inplace=True)
new df.notRepairedDamage.replace(('ja', 'nein'),('Yes', 'No'), inplace=
True)
new df.vehicleType.replace(('kleinwagen', 'cabrio', 'kombi', 'andere'),
 ('small car', 'convertible', 'combination', 'others'), inplace=True)
```

df['yearOfRegistration']=df['yearOfRegistration'].astype(int)

```
new df['price'].unique()
array(['18300', '9800', '1500', ..., '18429', '24895', '10985'],
      dtype=object)
new df['price'] = new df['price'].astype(int)
new df = new df[(new df.price \geq 100) & (new df.price \leq 150000)]
new df['fuelType'].fillna (value='not-declared', inplace=True)
new df['gearbox'].fillna (value='not-declared', inplace=True)
new df['notRepairedDamage'].fillna (value='not-declared', inplace=True)
new df[ 'vehicleType'].fillna (value='not-declared', inplace=True)
new df['model'].fillna (value='not-declared', inplace=True)
new df['kilometer']=new df['kilometer'].astype(int)
new df.info()
<class 'pandas.core.frame.DataFrame'>
Int64Index: 278363 entries, 1 to 371538
Data columns (total 13 columns):
     Column
                            Non-Null Count
                                              Dtype
      -----
     seller
                            278363 non-null object
 0
     offerType
                            278363 non-null object
                            278363 non-null int64
 2
     price
 3
     vehicleType
                            278363 non-null object
     yearOfRegistration 278363 non-null int64
 4
 5
     gearbox
                            278363 non-null object
 6
                            278363 non-null int64
     powerPS
 7
     model
                            278363 non-null object
     kilometer
                            278363 non-null int64
 8
     monthOfRegistration 278363 non-null object
 10 fuelType
                            278363 non-null object
                            278363 non-null object
 11 brand
 12 notRepairedDamage
                            278363 non-null object
dtypes: int64(4), object(9)
memory usage: 29.7+ MB
```

new df['price'].unique()

## new\_df.head()

new\_df.head()

	seller	offerType	price	vehicleType	yearOfRegistration	gearbox	powerPS	model	kilometer	monthOfRegistration	fuelType	brand	notRepairedDamage
1	privat	Angebot	18300	coupe	2011	manual	190	not-declared	125000	5	diesel	audi	Yes
2	privat	Angebot	9800	suv	2004	automatic	163	grand	125000	8	diesel	jeep	not-declared
3	privat	Angebot	1500	small car	2001	manual	75	golf	150000	6	petrol	volkswagen	No
4	privat	Angebot	3600	small car	2008	manual	69	fabia	90000	7	diesel	skoda	No
5	privat	Angebot	650	limousine	1995	manual	102	3er	150000	10	petrol	bmw	Yes