

	Pays	Superficie	Habitant	Densité
0	Russ	17098242	146781095	8.584572
1	Canada	998467	37560207	37.617875
2	USA	9826675	331883986	33.773783
3	Chine	9596961	1394112547	145.266043
4	Brésil	8514877	210301591	24.698136
5	Australie	774122	25105503	32.430939

Annexe 2(manipulation de date frame)

Print(date frame)	Retourne les données d'un dataframe	import pandas as ps df=ps.read_csv("pays.csv",sep=";") print (df)
Print(dataframe.head())	Retourne les 5 premières lignes	import pandas as ps df=ps.read_csv("pays.csv",sep=";") print (df.head())
Print(dataframe.tail())	Retourne les 5 dernières lignes	import pandas as ps df=ps.read_csv("pays.csv",sep=";") print (df.tail())
Print(dataframe.Nom_Colonne) ou bien print(dataframe["Nom_Colonne"])	Retourne les données d'une colonne	import pandas as ps df=ps.read_csv("pays.csv",sep=";") print(df.Pays)
Print(dataframe[["Nom_Col1","Nom_Col2"]])	Retourne les données de la col1 et de col2	import pandas as ps df=ps.read_csv("pays.csv",sep=";") print(df[["Pays","Habitant"]])
Print(dataframe.iloc[P_Initiale : P_Finale])	Retourne les données des lignes de la p_ initiale et la p_ finale - 1	import pandas as ps df=ps.read_csv("pays.csv",sep=";") print (df.iloc[0:3])
Print(dataframe.nom col [P_Initiale : P_Finale])	Retourne les données d'une colonne entre une p_ initiale et une p_ finale - 1	import pandas as ps df=ps.read_csv("pays.csv",sep=";") print (df.Pays[2:5])
Print(dataframe[dataframe ["nom colonne"] opérateur_comparaison valeur])	Affichage des données d'un dataframe selon une condition	import pandas as ps df=ps.read_csv("pays.csv",sep=";") print (df[df["Superficie"] == 9826675])
Print(dataframe[(dataframe ["Nom_Col"] Opérateur_Comparaison Valeur) Opérateur_Logique (dataframe ["Nom_Col"] Opérateur_Comparaison Valeur)])	Affichage des données d'un dataframe selon plusieurs conditions	print (df[(df["Superficie"] >= 9826675) (df["Superficie"] >= 1000000)])
Print(dataframe.info())	Affichage des informations concernant un dataframe	import pandas as ps df=ps.read_csv ("pays.csv" , sep= ";") print (df.info())

Print dataframe.columns.values)	Retourne les noms des colonnes sous forme d'un tableau	df=ps.read_csv("pays.csv",sep=";") print (df. Columns.values)
Print dataframe.dtypes)	Retourne les noms des colonnes suivies de leurs types	df=ps.read_csv("pays.csv",sep=";") print (df.dtypes)
Dataframe["Nom_Colonne"]=valeur	Ajouter une colonne	df=ps.read_csv("pays.csv",sep=";") df["Densité"]=df.Habitant/df.Superficie
df.rename(columns={"Colonne1": "Nouveau_Colonne1"},inplace=True/False)	renommer les libelles d'une colonne	df=ps.read_csv ("Pays.csv",sep=";") df.rename(columns={"Pays": 'Nom_Pays'},inplace=True)
del dataframe["Nom_Colonne"]	Supprimer une colonne	df=ps.read_csv ("pays.csv",sep=";") del df['Habitant']
Dataframe.loc[N° ligne, "Nom_colonne"]=Valeur	modifier le contenu d'une cellule	df=ps.read_csv("pays.csv",sep=";") df.loc[0, "Pays"] = "Russie"
Dataframe.loc[N° ligne]=[La liste des valeurs]	modifier le contenu d'une ligne	df=ps.read_csv("pays.csv",sep=";") df.loc[0] =["Russie" , 17098555,146781999]
dataframe["Nom_Col_Resultat"]=numpy.where(dataframe ["Nom_Col] opérateur_comparaison valeur, Valeur1_Si_vrai, Valeur2_si_Faux)	Remplissage d'une colonne suite à une condition simple	df=ps.read_csv("pays.csv",sep=";") df["Densité"]=df.Habitant/df.Superficie df["Observation"]=numpy.where(df["Densité"]>60,"Très dense", "Moyenne")
df=ps.read_csv("pays.csv",sep=";") df["Densité"]=df.Habitant/df.Superficie obser=[] for x in df["Densité"] : # parcourir les valeurs de la colonne densité valeur par valeur if x >=100 : # vérifier si la valeur de la densité >=100 obser.append("Très dense") else : if x >=50 : # vérifier si la valeur de la densité >=50 obser.append("Normale") else : if x >=20 : # vérifier si la valeur de la densité >=20 obser.append("Moyenne") else : obser.append("Faible") df["Observation"]=obser	Remplissage d'une colonne suite à plusieurs conditions	def calcul(x): if x >=100 : # vérifier si la valeur de la densité >=100 return "Très dense" else : if x >=50 : # vérifier si la valeur de la densité >=50 return "Normale" else : if x >=20 : # vérifier si la valeur de la densité >=20 return "Moyenne" else : return "Faible" df=ps.read_csv("pays.csv",sep=";") df["Densité"]=df.Habitant/df.Superficie df["Observation"]=df["Densité"].apply(calcul)

dataframe_resultat = pandas.concat([df1, df2, fdn], ignore_index = True)	Concaténation de plusieurs data frames	import pandas as ps df0=ps.read_csv("pays.csv",sep=";") df1=ps.read_csv("pays1.csv",sep=";") df_res=ps.concat([df0,df1],ignore_index = True)
Nouveau_dataframe= pandas.DataFrame([[La liste des valeurs]],columns=[La liste des colonnes]) dataframe= dataframe.append(Nouveau_dataframe ,sort=False, ignore_index=True/False)	Ajouter une ligne à la fin de data frame	import pandas as ps df=ps.read_csv("pays.csv",sep=";") df1=ps.DataFrame([["Tunisie",162155,11582075]],columns=["Pays","Superficie","Habitant"]) df=df.append(df1,sort=False,ignore_index=True)
	Ajouter une ligne à une position P	import pandas as ps df=ps.read_csv("pays.csv",sep=";") df0=df.iloc[0:p] df1=df.iloc[p:] df2=ps.DataFrame([["Tunisie",162155,11582075]],columns=["Pays","Superficie","Habitant"]) df0=df0.append(df2,sort=False,ignore_index=True) df=df0.append(df1,sort=False,ignore_index=True)
dataframe = dataframe.drop([indice_ligne])	Supprimer une ligne	import pandas as ps df=ps.read_csv("pays.csv",sep=";") df = df.drop(N° de ligne)
dataframe = dataframe. drop_duplicates()	supprimer les doublons d'un dataframe	import pandas as ps df=ps.read_csv("pays.csv",sep=";") df=df.drop_duplicates()
dataframe = dataframe.sort_values(by = [Liste des colonnes], ascending = [critère pour chaque colonne (True/False)])	trier les données d'un dataframe	import pandas as ps df=ps.read_csv("pays.csv",sep=";") df=df.sort_values(by = ["Superficie","Habitant"],ascending = [False,True])