	m_Identifier Ite	m_Weight Item	_Fat_Content It	tem_Visibility 0.016047	Item_Type Dairy	1tem_MRP Out 249.8092	tlet_Identifier Outlet_Esta	blishment_Yea 1999
2	DRC01 FDN15	5.92 17.50	Regular Low Fat	0.016760	Soft Drinks Meat Fruits and	48.2692 141.6180	OUT018 OUT049	2009 1999
tast	FDX07 NCD19	19.20 8.93	Regular Low Fat	0.000000	Fruits and Vegetables Household	182.0950 53.8614	OUT010 OUT013	1998 1987
	FDW58	m_Weight Item 20.750 8.300	Low Fat	0.007565	Snack Foods	107.8622	tlet_Identifier Outlet_Esta OUT049 OUT017	1999
	FDW14 NCN55 FDQ58	8.300 14.600 7.315	reg Low Fat Low Fat	0.038428 0.099575 0.015388	Others Snack Foods	87.3198 241.7538 155.0340	OUT017 OUT010 OUT017	2007 1998 2007
	FDY38	NaN	Regular	0.118599	Foods Dairy	234.2300	OUT027	1985
8523	n.shape 3, 12) .shape							
trai	n.isna().sum	()						
tem_ tem_ tem_ tem_ utle utle utle utle	Identifier Weight Fat_Content Visibility Type MRP et_Identifier et_Establishm et_Size et_Location_T et_Type Outlet_Sales e: int64	nent_Year 'ype	0 1463 0 0 0 0 0 0 0 0 2410 0					
tem_ tem_ tem_	.isna().sum(Identifier Weight Fat_Content Visibility)	0 976 0					
tem_ utle utle utle utle	Type MRP et_Identifier et_Establishm et_Size et_Location_T et_Type e: int64	nent_Year	0 0 0 0 1606 0					
e m ere si	oving miss	ing values oduct categorie se it fill in the m	s can have diff				y categories and find the	e medium we
	an_weights	train.group	oby(['Item_T	'ype'],as_i	ndex =Fals	e).agg({'Ite	m_Weight': np.mediar	n})
) 1 2	Baking Good Bread Breakfa Canne	ds 11.650 ds 10.600 sst 10.695						
4 5	Canne Dai Frozen Food ruits and Vegetable	ry 13.350 ds 12.850						
7 3 9	Hard Drinl Health and Hygier Househo Me.	ne 12.150 Id 13.150						
1 2 3	Othe Seafoo Snack Food	nrs 14.500 od 11.650 ds 13.150						
		ds 13.175	e(data=media	.n_weight。	values. c	olumns =['T+	em_Type','Avg_Item_W	Weight'll
	an_weights Item_Typ Baking Good	pe Avg_Item_We ds 1	ight 1.65	orynts.	CC		, Avg_Item_N	([۱
! ! !	Bread Breakfa Canne Dai	st 10 ed 1	10.6 0.695 2.15 3.35					
5 Fr	Frozen Food ruits and Vegetable Hard Drinl	ds 1 es ks	2.85 13.1 10.1					
	Health and Hygier Househo Me Othe	ld 1 at 1	2.153.152.3514.5					
} }	Seafoo Snack Food Soft Drinl	od 1 ds 1 ks	1.65 3.15 11.8					
			right = med	lian_weight	s, on = ['Item_Type']	, how='left')	
	n.head() m_Identifier Ite FDA15	m_Weight Item	_ Fat_Content It Low Fat	tem_Visibility	Item_Type Dairy	1tem_MRP Out 249.8092	tlet_Identifier Outlet_Esta	blishment_Ye 199
	DRC01 FDN15	5.92 17.50	Regular Low Fat		Soft Drinks Meat	48.2692 141.6180	OUT018 OUT049	200
	FDX07 NCD19	19.20 8.93	Regular Low Fat	0.000000	Fruits and Vegetables Household	182.0950 53.8614	OUT010 OUT013	199 198
	'Item_Type	tifier', 'It. ', 'Item_MRP	', 'Outlet I	dentifier'	,			
rai	'Outlet_Ty dtype='obje	rpe', 'Item_O	utlet_Sales'	, 'Avg_Ite	m_Weight'],	e x[1], axis=1)	
tem_	n.isna().sum Identifier Weight Fat Content	()	0 0 0					
tem ⁻	Visibility		0 0 0 0					
tem_tem_tem_tem_tem_utle	Type MRP t_Identifier t_Establishm t_Size t_Location_T	ment_Year	0 2410 0					
cem_ cem_ cem_ utle utle utle utle cem_ yg_I	Type MRP et_Identifier et_Establishm et_Size et_Location_T et_Type Outlet_Sales etem_Weight e: int64	ent_Year 'ype	2410 0 0 0 0	values				
tem_ tem_ tem_ tem_ utle utle utle utle utle rou rrai	Type MRP et_Identifier et_Establishm et_Size et_Location_T et_Type Outlet_Sales item_Weight e: int64 tlet size we use n['Outlet_Si v(['Medium',	the mode to fill ze'].unique() nan, 'High',	2410 0 0 0 0 in the missing		et)			
cem_cem_cem_cem_cem_utle cem_utle utle utle utle utle cem_rou rai	Type MRP et_Identifier et_Establishm et_Size et_Location_T et_Type Outlet_Sales tem_Weight e: int64 tlet size we use n['Outlet_Si n['Outlet_Si n['Outlet_Si ng'('Medium', ng'('Mediu	the mode to fill	2410 0 0 0 0 in the missing		et)			
cem_cem_cem_cem_cem_cem_cem_cem_utledutledutledutledutledutledutledutled	Type MRP et_Identifier et_Establishm et_Size et_Location_T et_Type Outlet_Sales Item_Weight e: int64 tlet size we use n['Outlet_Si n]'Outlet_Si n['Outlet_Si n]	the mode to fill ze'].unique() nan, 'High', ze'].value_co	2410 0 0 0 0 in the missing 'Small'], d	ltype=objec		'Medium' if	pd.isna(x) else x)	
cem_cem_cem_utle cem_utle utle utle utle utle utle utle utle	Type MRP et_Identifier et_Establishm et_Size et_Location_T et_Type Outlet_Sales etem_Weight e: int64 tlet size we use n['Outlet_Si '(['Medium', n['Outlet_Si cting errone n['Item_Fat_ '(['Low Fat',	the mode to fill ze'].unique() nan, 'High', ze'].value_co e, dtype: int ze'] = train ous data Content'].un: 'Regular',	2410 0 0 0 in the missing 'Small'], of ounts() 64 ['Outlet_Siz	dtype=object e'].apply(lambda x:	object)		
cem_cem_cem_cem_cem_utledutledutledutledutledutledutledutled	Type MRP et_Identifier et_Establishm et_Size et_Location_T et_Type Outlet_Sales Etem_Weight e: int64 tlet size we use n['Outlet_Si '(['Medium', n['Outlet_Si cting errone n['Item_Fat_ '(['Low Fat', n['Item_Fat_ '(['Low Fat', n['Item_Fat_ '(['Item_Fat_ '(['Item_	the mode to fill ze'].unique() nan, 'High', ze'].value_co e, dtype: int ze'] = train ous data Content'].un: 'Regular', Content'] = t	2410 0 0 0 in the missing 'Small'], of ounts() 64 ['Outlet_Siz ique() 'low fat', ' crain['Item_	dtype=object e'].apply(LF', 'reg' Fat_Conten	lambda x:	object)	<pre>pd.isna(x) else x)</pre> Low Fat', 'Regular':	'Regular'
tem_ tem_ tem_ tem_ tem_ tem_ tem_ tem_	Type MRP et_Identifier et_Establishm et_Size et_Location_T et_Type Outlet_Sales item_Weight e: int64 tlet size we use n['Outlet_Si '(['Medium', n['Outlet_Si cting errone n['Item_Fat_ '(['Low Fat', n['Item_Fat_ '(['Low Fat', n.drop(['Item_Fat_ '(['Low Fat', n.drop(['Low Fat'	the mode to fill ze'].unique() nan, 'High', ze'].value_co a, dtype: int ze'] = train ous data Content'].un: 'Regular', Content'] = t Content'].un: 'Regular'], m_Identifier	2410 0 0 0 0 in the missing 'Small'], d ounts() 64 ['Outlet_Siz ique() 'low fat', ' crain['Item_ ique() dtype=object ','Outlet_Id	e'].apply(LF', 'reg' Fat_Conten	<pre>lambda x:], dtype= t'].map({</pre>	object) 'Low Fat': '		'Regular'
tem_tem_tem_tem_tem_tem_tem_tem_tem_tem_	Type MRP et_Identifier et_Establishm et_Size et_Location_T et_Type Outlet_Sales item_Weight e: int64 tlet size we use n['Outlet_Si '(['Medium', n['Outlet_Si '(['Medium', n['Outlet_Si '(['Medium', n['Outlet_Si '(['Low Fat', n['Item_Fat_ '(['Supermark 'Supermark 'Supermark	the mode to fill ze'].unique() nan, 'High', ze'].value_co e, dtype: int ze'] = train ous data Content'].un: 'Regular', Content'] = t Content'].un: 'Regular'], m_Identifier pe'].unique() tet Type1', 'tet Type3'], tet Type3'],	2410 0 0 0 0 in the missing 'Small'], of ounts() 64 ['Outlet_Siz ique() 'low fat', ' crain['Item_ ique() dtype=object ','Outlet_Id Supermarket dtype=object	dtype=object de'].apply(LF', 'reg' Fat_Conten ct) Type2', 'G	<pre>lambda x:], dtype= t'].map({ 'Avg_Item_ Grocery St</pre>	object) 'Low Fat': ': _Weight'],ax ore',	Low Fat', 'Regular': is=1, inplace=True)	
tem_tem_tem_tem_tem_tem_tem_tem_tem_tem_	Type MRP et_Identifier et_Establishm et_Size et_Location_T et_Type Outlet_Sales item_Weight e: int64 tlet size we use n['Outlet_Si '(['Medium', n['Outlet_Si '(['Medium', n['Outlet_Si '(['Medium', n['Outlet_Si '(['Low Fat', n['Item_Fat_ '(['Supermark 'Supermark 'Supermark	the mode to fill ze'].unique() nan, 'High', ze'].value_co e, dtype: int ze'] = train ous data Content'].un: 'Regular', Content'] = t Content'].un: 'Regular'], m_Identifier pe'].unique() tet Type1', 'tet Type3'], tet Type3'],	2410 0 0 0 0 in the missing 'Small'], of ounts() 64 ['Outlet_Siz ique() 'low fat', ' crain['Item_ ique() dtype=object ','Outlet_Id Supermarket dtype=object	dtype=object de'].apply(LF', 'reg' Fat_Content Type2', 'G t_Content'	<pre>lambda x:], dtype= t'].map({ 'Avg_Item_</pre>	object) 'Low Fat': ': _Weight'],ax ore',	Low Fat','Regular':	
tem_tem_tem_tem_tem_tem_tem_tem_tem_tem_	Type MRP et_Identifier et_Establishm et_Size et_Location_T et_Type Outlet_Sales item_Weight e: int64 tlet size we use n['Outlet_Si "(['Medium', n['Outlet_Si "(['Medium', n['Outlet_Si "(['Low Fat', n['Item_Fat_ "(['Low Fat', n['Item_Fat_ "(['Low Fat', n.drop(['Ite n['Outlet_Ty "(['Supermark n = pd.get_d n.head()	the mode to fill ze'].unique() nan, 'High', ze'].value_co c, dtype: int ze'] = train ous data Content'].un: 'Regular', Content'] = t Content'].un: 'Regular'], m_Identifier pe'].unique() tet Type3'], ummies (column	in the missing 'Small'], of ounts() fique() 'low fat', ' crain['Item_ dque() dtype=object ','Outlet_Id Supermarket dtype=object self: Item_Fat drop_first=	dtype=object de'].apply(LF', 'reg' Fat_Conten tt) Type2', 'G t_Content' True, data=	<pre>lambda x:], dtype= t'].map({ 'Avg_Item_ rocery St ,'Item_Type</pre>	object) 'Low Fat': ': _Weight'],ax ore', pe','Outlet_	Low Fat', 'Regular': is=1, inplace=True)	ion_Type',
tem_ tem_ tem_ tem_ tem_ tem_ tem_ tem_	Type MRP et_Identifier et_Establishm et_Size et_Location_T et_Type Outlet_Sales item_Weight e: int64 tlet size we use n['Outlet_Si "(['Medium', n['Outlet_Si "(['Medium', n['Outlet_Si "(['Medium', n['Item_Fat_ "(['Low Fat', n['Item_Fat_ "(['Low Fat', n.drop(['Item_Fat_ "(['Low Fat', n.drop(['Item_Fat_ "(['Supermark	the mode to fill ze'].unique() nan, 'High', ze'].value_co c, dtype: int ze'] = train ous data Content'].un: 'Regular', Content'] = t Content'].un: 'Regular'], m_Identifier pe'].unique() tet Type3'], ummies (column Let Type3'], ummies (column 0.016047 249. 0.019278 48. 0.016760 141. 0.0000000 182.	in the missing 'Small'], of ounts() ique() 'low fat', ' crain['Item_ ique() dtype=object y,'Outlet_Id Supermarket dtype=object self' Item_Fat drop_first= MRP Outlet_Es 8092	dtype=object de'].apply(LF', 'reg' Fat_Conten Type2', 'G t_Content' True, data= tablishment_N 1 2 1 1	<pre>lambda x:], dtype= t'].map({ 'Avg_Item_ Grocery St ,'Item_Typ train) /ear Item_Of 999</pre>	object) 'Low Fat': ': _Weight'],ax ore', pe','Outlet_ utlet_Sales Item 3735.1380	Low Fat', 'Regular': is=1, inplace=True) Size', 'Outlet_Locati n_Fat_Content_Regular Ite 0	ion_Type',
tem_tem_tem_tem_tem_tem_tem_tem_tem_tem_	Type MRP et_Identifier et_Establishm et_Size et_Location_T et_Type Outlet_Sales item_Weight e: int64 tlet size we use n['Outlet_Si '(['Medium', n['Outlet_Si '(['Medium', n['Outlet_Si cting errone n['Item_Fat_ '(['Low Fat', n['Item_Fat_ '(['Low Fat', n.drop(['Item n['Item_Fat_ '(['Low Fat', n.drop(['Item n['Outlet_Ty '(['Supermark n = pd.get_d n.head() m_Weight Item 9.30 5.92 17.50 19.20 8.93 × 28 columns he establishmer	the mode to fill ze'].unique() nan, 'High', ze'].value_co a, dtype: int ze'] = train ous data Content'].un: 'Regular', Content'] = t Content'].un: 'Regular'], m_Identifier pe'].unique() tet Type3'], tet Type3'], ummies (column 0.016047 249. 0.016760 141. 0.0000000 182. 0.0000000 53.	in the missing 'Small'], counts() ounts() fique() 'low fat', 'crain['Item_ ique() dtype=object ','Outlet_Id Supermarket dtype=object s=['Item_Fa drop_first= drop_first= MRP Outlet_Es 8092 2692 6180 0950 8614	dtype=object detype=object detype=object detype=object detype=object detype=object fetype=object fetype=	lambda x:], dtype= t'].map({ 'Avg_Item_ fracery St ,'Item_Typ train) /ear Item_Or 999 999 998 987	object) 'Low Fat': '! _Weight'],ax ore', oe','Outlet_ attlet_Sales Item 3735.1380 443.4228 2097.2700 732.3800 994.7052	Low Fat', 'Regular': is=1, inplace=True) Size', 'Outlet_Locati 0 1 0 1 0 1 0 1 0 1 0	ion_Type',
tem_tem_tem_tem_tem_tem_tem_tem_tem_tem_	Type MRP et_Identifier et_Establishm et_Size et_Location_T et_Type Outlet_Sales item_Weight e: int64 tlet size we use n['Outlet_Si "(['Medium', n['Outlet_Si "(['Medium', n['Outlet_Si "(['Medium', n['Item_Fat_ "(['Low Fat', n['Item_Fat_ "(['Low Fat', n['Item_Fat_ "(['Low Fat', n.drop(['Item_Ned] n.head() m_Weight Item_ 9.30 5.92 17.50 19.20 8.93 × 28 columns he establishmer n['Outlet_Ag "(Outlet_Ag "(Outlet_Ag	the mode to fill ze'].unique() nan, 'High', ze'].value_co a, dtype: int ze'] = train ous data Content'].un: 'Regular', Content'] = t Content'].un: 'Regular'], m_Identifier pe'].unique() tet Type1', 't tet Type3'], ummies(column	in the missing 'Small'], counts() 'Small'], counts() 'Quetalet_Size 'que() 'low fat', 'crain['Item_ 'que() 'dtype=object ','Outlet_Id Supermarket dtype=object 'serial ('Item_Fart) adrop_first= 'Outlet_Esta	dtype=object dtype=object de'].apply(LF', 'reg' Fat_Content Type2', 'G t_Content' True, data= tablishment_ te, we use it to blishment_	lambda x:], dtype= t'].map({ 'Avg_Item_ 'Avg_Item_ train) fear Item_Or 999 009 999 988 987 o find the ac Year'].app	object) 'Low Fat': '! _Weight'],ax ore', oe','Outlet_ attlet_Sales Item 3735.1380 443.4228 2097.2700 732.3800 994.7052 ge of the stores oly(lambda x	Low Fat', 'Regular': is=1, inplace=True) Size', 'Outlet_Locati 0 1 0 1 0 1 0 1 0 1 0	ion_Type',
tem_tem_tem_tem_tem_tem_tem_tem_tem_tem_	Type MRP et_Identifier et_Establishmet size et_Location_T coutlet_Sales et_Type Outlet_Sales ftem_Weight et int64 tlet size we use n['Outlet_Si m	the mode to fill ze'].unique() nan, 'High', ze'].value_co e, dtype: int ze'] = train ous data Content'].un: 'Regular', Content'] = f Content'].un: 'Regular'], m_Identifier pe'].unique() tet Type3'], ummies (column Let Type3'], ummies (column ous data content'] = f content'].un: let Type1', ' tet Type3'], ummies (column Let Type3'), ummies	in the missing 'Small'], depends of the state of the sta	dtype=object dtype=object e'].apply(LF', 'reg' Fat_Content Type2', 'G t_Content' True, data= fe, we use it t blishment_ , axis=1, i	lambda x:], dtype= t'].map({ 'Avg_Item_ Grocery St ,'Item_Typ train) 'ear Item_Or 999 999 998 987 ofind the ac Year'].app nplace=Train	object) 'Low Fat': ': _Weight'],ax ore', oe','Outlet_ atlet_Sales Item 3735.1380 443.4228 2097.2700 732.3800 994.7052 ge of the stores oly(lambda x ue)	Low Fat', 'Regular': is=1, inplace=True) Size', 'Outlet_Locati n_Fat_Content_Regular	ion_Type',
tem_tem_tem_tem_tem_tem_tem_tem_tem_tem_	Type MRP et_Identifier et_Establishm et_Size et_Location_T coutlet_Sales ftem_Weight e: int64 tlet size we use n['Outlet_Si "(['Medium', n['Outlet_Si "(['Medium', n['Outlet_Si "(['Low Fat', n['Item_Fat_ "(['Low Fat', n['Item_Fat_ "(['Low Fat', n.drop(['Itemoneter) n['Outlet_Ty "(['Supermark	the mode to fill ze'].unique() nan, 'High', ze'].value_co a, dtype: int ze'] = train ous data Content'].un: 'Regular', Content'] = f Content'].un: 'Regular'], "Type3'], at Type3'], at Type3	2410 0 0 0 in the missing 'Small'], depends of the control of the	dtype=object de'].apply(LF', 'reg' Fat_Conten Type2', 'G t_Content' True, data= fe, we use it to blishment_ ,axis=1, i	lambda x:], dtype= t'].map({ 'Avg_Item_ Grocery St ,'Item_Typ train) 'ear Item_Or 999 999 998 987 ofind the ac Year'].app nplace=Train	object) 'Low Fat': ': _Weight'],ax ore', oe','Outlet_ arange of the stores oly(lambda x ue)	Low Fat', 'Regular': is=1, inplace=True) Size', 'Outlet_Locati n_Fat_Content_Regular Ite 0 1 0 1 0 1 0 1 0 1 1 1 1	ion_Type',
tem_ tem_ tem_ tem_ tem_ tem_ tem_ tem_	Type MRP et_Identifier et_Establishment_size et_Location_Tet_Type Outlet_Sales et_Location_Tet_Type Outlet_Sales et_House et_Inter_Sales et_House et_Inter_Sales et_Meight et int64 tlet size we use n['Outlet_Si cting errone n['Item_Fat '(['Low Fat', n['Item_Fat '(['Low Fat', n['Item_Fat '(['Low Fat', n.drop(['Item_Fat '(['Low Fat', n.drop(['Item_Pat '(['Supermark n = pd.get_d n.head() m_Weight Item_ 9.30 5.92 17.50 19.20 8.93 x 28 columns he establishmer n['Outlet_Ag n.drop(['Out n.head() m_Weight Item_ 9.30 5.92 17.50 19.20 8.93 x 28 columns he establishmer n['Outlet_Ag n.drop(['Out n.head())	the mode to fill ze'].unique() nan, 'High', ze'].value_co a, dtype: int ze'] = train ous data Content'].un: 'Regular', Content'] = t Content'].un: 'Regular'], m_Identifier pe'].unique() tet Type1', ' tet Type3'], ' ummies(column ous data Content'].un: let Type1', ' tet Type3'], ' tet Type3'], ' tet Type3'], ' ummies(column ous data content'].un: let_Establish disnot an impon e'] = train[let_Establish ous data content'].un: let_Establish ous data disnot an impon e'] = train[let_Establish ous data disnot an impon e'] = train[let_Establish ous data disnot an impon e'] = train[let_Establish ous data disnot an impon e'] = train[let_Establish ous data	in the missing 'Small'], depends of the missing 'Small'], depends of the missing 'Small'], depends of the missing 'Que() 'Iow fat', 'Carain['Item_ 'Que() ','Outlet_Identify and the missing MRP Outlet_Estantify and the missing MRP Outlet_Estantify and the missing MRP Outlet_Estantify and the missing MRP Item_Outlet_Estantify and the missing	dtype=object dtype=object dtype=object e'].apply(LF', 'reg' Fat_Content Type2', 'G t_Content' True, data= te, we use it t blishment_ , axis=1, i delet_Sales telet 1335.1380 443.4228 1097.2700	lambda x:], dtype= t'].map({ 'Avg_Item_ Grocery St ,'Item_Typ train) 'ear Item_Or 999 999 998 987 ofind the ac Year'].app nplace=Train	object) 'Low Fat': ': _Weight'],ax ore', pe','Outlet_ array a	Low Fat', 'Regular': is=1, inplace=True) Size', 'Outlet_Locati n_Fat_Content_Regular Ite 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 0	ion_Type', em_Type_Brea Demotion_Type_O O O O
tem_tem_tem_tem_tem_tem_tem_tem_tem_tem_	Type MRP tt_Identifier et_Identifier et_Establishm et_Size et_Location_T et_Type Outlet_Sales etint64 tlet size we use n['Outlet_Si '(['Medium', n['Outlet_Si '(['Medium', n['Outlet_Si cting errone n['Item_Fat_ '(['Low Fat', n['Item_Fat_ '(['Low Fat', n['Item_Fat_ '(['Low Fat', n.drop(['Item	the mode to fill ze'].unique() nan, 'High', ze'].value_co a, dtype: int ze'] = train ous data Content'].un: 'Regular', Content'] = t Content'].un: 'Regular'], m_Identifier pe'].unique() tet Type1', ' tet Type3'], ' ummies(column ous data Content'].un: let Type1', ' tet Type3'], ' tet Type3'], ' tet Type3'], ' ummies(column ous data content'].un: let_Establish disnot an impon e'] = train[let_Establish ous data content'].un: let_Establish ous data disnot an impon e'] = train[let_Establish ous data disnot an impon e'] = train[let_Establish ous data disnot an impon e'] = train[let_Establish ous data disnot an impon e'] = train[let_Establish ous data	in the missing 'Small'], counts() ounts() fique() 'low fat', 'carain['Item_ ique() dtype=object catype=object s=['Item_Fadrop_first= drop_first= MRP Outlet_Es 8092 2692 6180 0950 8614 Outlet_Esta ment_Year'] MRP Item_Out 8092 3 2692 6180 20950 8614	dtype=object dtype=object de'].apply(LF', 'reg' Fat_Content Type2', 'G t_Content' true, data= fe, we use it t blishment_ axis=1, i fe, we use it t ce, we use it t data= for axis=1, i fo	lambda x:], dtype= t'].map({ 'Avg_Item_ Grocery St ,'Item_Typ train) 'ear Item_Or 999 009 999 998 987 ofind the ac Year'].app nplace=Train m_Fat_Conter	object) 'Low Fat': ': _Weight'],ax ore', oe','Outlet_ atlet_Sales Item 3735.1380 443.4228 2097.2700 732.3800 994.7052 ge of the stores oly(lambda x ue) ot_Regular Item o 1 0 1	Low Fat', 'Regular': is=1, inplace=True) Size', 'Outlet_Location n_Fat_Content_Regular	Em_Type_Breat O O O
tem_ tem_ tem_ tem_ tem_ tem_ tem_ tem_	Type MRP Et_Identifier Et_Establishmet_Size Et_Location_T Et_Type Outlet_Sales Etem_Weight Et int64 Itlet size we use I ('Outlet_Si I ('Medium', I ('Medium', I ('Outlet_Si I ('Item_Fat_ I ('I	the mode to fill ze'].unique() nan, 'High', ze'].value_co data Content'].un: 'Regular', Content'].un: 'Regular'], m_Identifier pe'].unique() tet Type3'], ummies(column Let Type3'], ummies(column pe'] = train[let_Establis] List Establis]	in the missing 'Small'], of ounts() 'Small'], of ounts() 'Small'], of ounts() 'Aque() 'Iow fat', ' crain['Item_ dque() ','Outlet_Id Supermarket chype=object as=['Item_Fa drop_first= outlet_Esta outlet_Esta outlet_Esta outlet_Esta outlet_Esta outlet_Sales c_Sales']] MRP Item_Outlet outlet_Sales c_Sales']]	dtype=object dtype=object dtype=object de'].apply(LF', 'reg' Fat_Content Type2', 'd t_Content' true, data= det, data= det, data= fee, we use it to blishment_ , axis=1, i det, data= det, d	lambda x:], dtype= t'].map({ 'Avg_Item_ Grocery St ,'Item_Typ train) /ear Item_Of 999 999 998 987 ofind the acc Year'].app nplace=Train nplace=Train	object) 'Low Fat': ': _Weight'],ax ore', pe','Outlet_ 3735.1380 443.4228 2097.2700 732.3800 994.7052 ge of the stores ply(lambda x ue) ot_Regular tem ot_ 1 0 1 0 1 0	Low Fat', 'Regular': is=1, inplace=True) Size', 'Outlet_Location n_Fat_Content_Regular	Breakfast It 0 0 0 0
tem_ tem_ tem_ tem_ tem_ tem_ tem_ tem_	Type MRP St_Identifier st_Establishments_size st_Location_Tet_Type Outlet_Sales stem_Weight sint64 tlet size we use n['Outlet_Si "(['Medium', n['Outlet_Si "(['Medium', n['Outlet_Si "(['Low Fat', n['Item_Fat_ "(['Low Fat', n['Item_Fat_ "(['Low Fat', n.drop(['Item n['Outlet_Ty "(['Supermark	the mode to fill ze'].unique() nan, 'High', ze'].value_co , dtype: int ze'] = train ous data Content'].un: 'Regular', Content'] = f Content'].un: 'Regular'], m_Identifier pe'].unique() let Type3'], let Type1', let Type1', let Type3'], let Type3'], let Type3'], let Type3'], let Type3'], let Type3'], let Type1', let Type1', let Type3'], let Type1', le	in the missing 'Small'], do o o in the missing 'Small'], do o o o in the missing 'Small'], do o o o o o o o in the missing 'Small'], do o o o o o o o o o o o o	dtype=object dtype=object dtype=object de'].apply(LF', 'reg' Fat_Content Type2', 'd t_Content' true, data= det, data= det, data= fee, we use it to blishment_ , axis=1, i det, data= det, d	lambda x:], dtype= t'].map({ 'Avg_Item_ Grocery St ,'Item_Typ train) /ear Item_Or 999 999 999 998 987 ofind the ac Year'].app nplace=Tru nplace=Tru nplace=Tru 1	object) Low Fat': '! Weight'], ax ore', pe', 'Outlet_ utlet_Sales Item 3735.1380 443.4228 2097.2700 732.3800 994.7052 ge of the stores oly(lambda x ue) pe_Breads Item 0 1 0 1 0 0 1 0 0 0 0 0	Low Fat', 'Regular': is=1, inplace=True) Size', 'Outlet_Location n_Fat_Content_Regular	Breakfast It O O O O O O O O O O O O O
tem_tem_tem_tem_tem_tem_tem_tem_tem_tem_	Type MRP St_Identifier St_Establishm St_Size St_Location_Tellocat	the mode to fill ze'].unique() nan, 'High', ze'].value_co , dtype: int ze'] = train ous data Content'].un: 'Regular', Content'] = f Content'].un: 'Regular'], m_Identifier pe'].unique() let Type3'], let Ty	in the missing 'Small'], or ounts() 'Small'], or ounts() 'Small'], or ounts() 'Algue() 'Iow fat', ' crain['Item_ dque() dtype=object ','Outlet_Id Supermarket dtype=object se['Item_Fate drop_first= MRP Outlet_Esta drop_first= MRP Item_Out 8092 2692 6180 0950 8614 MRP Item_Out 8092 3 2692 6180 20950 8614	dtype=object dtype=object e'].apply(LF', 'reg' Fat_Content Type2', 'G t_Content' True, data= det, we use it t blishment_ , axis=1, i fe, we use it t blishment_ 735.1380 443.4228 732.3800 7994.7052	lambda x:], dtype= t'].map({ 'Avg_Item_ Grocery St ,'Item_Typ train) /ear Item_Or 999 999 998 987 ofind the ag Year'].app nplace=Train nplace=Train nplace=Train ofine the age The conternation of the age The content of t	object) 'Low Fat': ': _Weight'],ax ore', oe','Outlet_ atlet_Sales Item 3735.1380 443.4228 2097.2700 732.3800 994.7052 ge of the stores oly(lambda x ue) pe_Breads Item 0 1 0 1 0 1 0 1 0 0 1 0 0	Low Fat', 'Regular': is=1, inplace=True) Size', 'Outlet_Locati n_Fat_Content_Regular Ite 0 1 0 1 0 1 0 1 0 1 Type_Breads Item_Type_ 0 0 0 0 0 0 0 0 0 0	Breakfast It 0 0 0 0 0 0 0 0 0 0 0 0
ows crai crai	Type MRP MRP St_Identifier St_Establishment St_Size St_Location_T St_Type Outlet_Sales Stem_Weight Stem_Weight Stem_Weight Stem_Weight Size We use n['Outlet_Si (''Medium', n['Outlet_Si am 2793 2388 932 Outlet_Size n['Outlet_Si cting errone n['Item_Fat_ ('(''Low Fat', n['Item_Fat_ ('(''Low Fat', n.drop(['Item_Fat_ ''(''Supermark n = pd.get_d n.head() m_Weight Item_ 9.30 5.92 17.50 19.20 8.93 x 28 columns he establishment n['Outlet_Ag n.drop(['Out n.head() m_Weight Item_ 9.30 5.92 17.50 19.20 8.93 x 28 columns he establishment n['Outlet_Ag n.drop(['Out n.head() m_Weight Item_ 9.30 5.92 17.50 19.20 8.93 x 28 columns sain = train.	the mode to fill ze'].unique() nan, 'High', ze'].value_co nan, 'High', ze'].value_co nan, 'High', ze'].value_co content'].un: 'Regular', Content'].un: 'Regular'], "Identifier pe'].unique() et Type1', ' tet Type3'], ' ummies(column Let Type1', ' tet Type3'], ' tet Type3'], ' tet Type3'], ' tet Type3'], ' content'] Let Type1', ' tet Type1', ' tet Type3'], ' tet Type1', ' tet Type3'], '	in the missing 'Small'], d ounts() 'Small'], d ounts() 'Author Size aque() 'Iow fat', ' crain['Item_ aque() dtype=object s=['Item_Father Size MRP Outlet_Est adrop_first= MRP Item_Out 8092 2692 6180 0950 8614 MRP Item_Out 8092 32692 6180 0950 8614 MRP Item_Father Size MRP Item_Size MRP Outlet_Est MRP Outlet_Est MRP Item_Out 8092 308614	dtype=object dtype=object e'].apply(LF', 'reg' Fat_Content Type2', 'G t_Content' True, data= det, we use it t blishment_ , axis=1, i fe, we use it t blishment_ 735.1380 443.4228 732.3800 7994.7052	lambda x:], dtype= t'].map({ 'Avg_Item_ Grocery St ,'Item_Typ train) /ear Item_Or 999 999 999 998 987 ofind the ac Year'].app nplace=Tru nplace=Tru nplace=Tru 1	object) Low Fat': '! Weight'], ax ore', pe', 'Outlet_ utlet_Sales Item 3735.1380 443.4228 2097.2700 732.3800 994.7052 ge of the stores oly(lambda x ue) pe_Breads Item 0 1 0 1 0 0 1 0 0 0 0 0	Low Fat', 'Regular': is=1, inplace=True) Size', 'Outlet_Location n_Fat_Content_Regular	Breakfast I O O O O O O O O O O O O O
tem_tem_tem_tem_tem_tem_tem_tem_tem_tem_	Type MRP tt_Establishments_size_stablishments_size_stablishments_size_we use n['Outlet_Sales_stem_Weight sint64 tlet size we use n['Outlet_Si n['Outlet_Si n['Outlet_Si n['Outlet_Si ng 2793 2388 932 Outlet_Size n['Outlet_Si n['Item_Fat n['Outlet_Ty, n.drop(['Item_n] n.drop(['Outlet_Ag	the mode to fill ze'].unique() nan, 'High', ze'].value_co a, dtype: int ze'] = train ous data Content'].un: 'Regular', Content'] = i content'].un: 'Regular'], m_Identifier pe'].unique() tet Type3'], ummies(column	in the missing 'Small'], do outlet_Siz ique() 'low fat', ' crain['Item_ ique() dtype=object seque() dtype=object seque() MRP Outlet_Est adrop_first= MRP Outlet_Est adrop_first= MRP Item_Out 8092 2692 6180 0950 8614 MRP Item_Out 8092 3 2692 6180 20950 8614	dtype=object de'].apply(LF', 'reg' Fat_Content t_Content' t_Content' te, we use it the blishment , axis=1, i det_Sales tel 735.1380 443.4228 1097.2700 732.3800 994.7052	lambda x:], dtype= t'].map({ 'Avg_Item_ Grocery St ,'Item_Typ train) /ear Item_Or 999 999 999 998 987 ofind the ac Year'].app nplace=Tru nplace=Tru nplace=Tru 1	object) Low Fat': '! Weight'], ax ore', pe', 'Outlet_ utlet_Sales Item 3735.1380 443.4228 2097.2700 732.3800 994.7052 ge of the stores oly(lambda x ue) pe_Breads Item 0 1 0 1 0 0 1 0 0 0 0 0	Low Fat', 'Regular': is=1, inplace=True) Size', 'Outlet_Location n_Fat_Content_Regular	Breakfast It 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0