

Name:

1) ARYAK BHODKE (J005)

2) SAKSHEE THAKKAR (J056)

3) PRAKHAR BANSAL (J069)

4) KRISHNA PATEL (J227)

MBA (TECH.), DIV-I

## Business Vizualization IV

### Indian Cuisine Analysis

Indian cuisine consists of a variety of regional and traditional cuisines native to the Indian subcontinent. Given the diversity in soil, climate, culture, ethnic groups, and occupations, these cuisines vary substantially and use locally available spices, herbs, vegetables, and fruits. Indian food is also heavily influenced by religion, in particular Hinduism, cultural choices and traditions.

This dataset consists of information about various Indian dishes, their ingredients, their place of origin, etc.

```
In [34]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
%matplotlib inline
```

### 1. Read the indianfood1.csv file into a DataFrame.

```
In [35]: df=pd.read_csv('indianfood1.csv')
df.head(10)
```

```
Out[35]:
```

	name	ingredients	diet	prep_time	cook_time	flavor_profile	course	state	region
0	Balu shahi	Maida flour, yogurt, oil, sugar	vegetarian	45	25	sweet	dessert	West Bengal	East
1	Boondi	Gram flour, ghee, sugar	vegetarian	80	30	sweet	dessert	Rajasthan	West
2	Gajar ka halwa	Carrots, milk, sugar, ghee, cashews, raisins	vegetarian	15	60	sweet	dessert	Punjab	North
3	Ghevar	Flour, ghee, kewra, milk, clarified butter, su...	vegetarian	15	30	sweet	dessert	Rajasthan	West
4	Gulab jamun	Milk powder, plain flour, baking powder, ghee,...	vegetarian	15	40	sweet	dessert	West Bengal	East
5	Imarti	Sugar syrup, lentil flour	vegetarian	10	50	sweet	dessert	West Bengal	East
6	Jalebi	Maida, corn flour, baking soda, vinegar, curd,...	vegetarian	10	50	sweet	dessert	Uttar Pradesh	North
7	Kaju katli	Cashews, ghee, cardamom, sugar	vegetarian	10	20	sweet	dessert	-1	-1

	name	ingredients	diet	prep_time	cook_time	flavor_profile	course	state	region
8	Kalakand	Milk, cottage cheese, sugar	vegetarian	20	30	sweet	dessert	West Bengal	East
9	Kheer	Milk, rice, sugar, dried fruits	vegetarian	10	40	sweet	dessert	-1	-1

In [36]: `df.shape`

Out[36]: (255, 9)

In [37]: `df.isnull().count()`

Out[37]:

name	255
ingredients	255
diet	255
prep_time	255
cook_time	255
flavor_profile	255
course	255
state	255
region	255

dtype: int64

## 2. How many total Indian dishes are there?

In [38]: `df['name'].nunique()`

Out[38]: 255

## 3. How many different set of features are there in every Indian dish?

In [39]: `df.columns`

Out[39]: Index(['name', 'ingredients', 'diet', 'prep\_time', 'cook\_time', 'flavor\_profile', 'course', 'state', 'region'], dtype='object')

## 4. Using Describe function, view the basic statistics of all columns. What Inference you can make out form that?

In [40]: `df.describe()`

Out[40]:

	prep_time	cook_time
count	255.000000	255.000000
mean	31.105882	34.529412
std	72.554409	48.265650
min	-1.000000	-1.000000
25%	10.000000	20.000000
50%	10.000000	30.000000
75%	20.000000	40.000000

	prep_time	cook_time
max	500.000000	720.000000

Inference: We observe that the mean of cooking time is greater than cooking time.

## 5. Are there any missing values in the dataset? If Yes, replace the missing values with the NaN values.

```
In [41]: df = df.replace(-1, np.nan)
df = df.replace('-1', np.nan)
df.head(10)
```

	name	ingredients	diet	prep_time	cook_time	flavor_profile	course	state	region
0	Balu shahi	Maida flour, yogurt, oil, sugar	vegetarian	45.0	25.0	sweet	dessert	West Bengal	East
1	Boondi	Gram flour, ghee, sugar	vegetarian	80.0	30.0	sweet	dessert	Rajasthan	West
2	Gajar ka halwa	Carrots, milk, sugar, ghee, cashews, raisins	vegetarian	15.0	60.0	sweet	dessert	Punjab	North
3	Ghevar	Flour, ghee, kewra, milk, clarified butter, su...	vegetarian	15.0	30.0	sweet	dessert	Rajasthan	West
4	Gulab jamun	Milk powder, plain flour, baking powder, ghee,...	vegetarian	15.0	40.0	sweet	dessert	West Bengal	East
5	Imarti	Sugar syrup, lentil flour	vegetarian	10.0	50.0	sweet	dessert	West Bengal	East
6	Jalebi	Maida, corn flour, baking soda, vinegar, curd,...	vegetarian	10.0	50.0	sweet	dessert	Uttar Pradesh	North
7	Kaju katli	Cashews, ghee, cardamom, sugar	vegetarian	10.0	20.0	sweet	dessert	NaN	NaN
8	Kalakand	Milk, cottage cheese, sugar	vegetarian	20.0	30.0	sweet	dessert	West Bengal	East
9	Kheer	Milk, rice, sugar, dried fruits	vegetarian	10.0	40.0	sweet	dessert	NaN	NaN

```
In [42]: df.isnull().sum()
```

```
Out[42]: name                0
ingredients              0
diet                    0
prep_time               30
cook_time               28
flavor_profile          29
course                  0
state                   24
region                  14
dtype: int64
```

```
In [43]: df['prep_time'].fillna(df.prep_time.median(), inplace=True)
```

## 6. How many numeric features and categorical features are there in the dataset?

```
In [44]: df.dtypes
```

```
Out[44]: name                object
ingredients            object
diet                   object
prep_time              float64
cook_time              float64
flavor_profile         object
course                 object
state                  object
region                 object
dtype: object
```

Out of these 9 columns: 2 are numeric features while 6 are categorical features

## 7. Added a new column in the dataset that will count the number of ingredients from the ingredients column for each dish.

```
In [45]: df['No_of_ingredients']=df.ingredients.apply(lambda x:len(x.split(',')))
```

```
In [46]: df.head(10)
```

Out[46]:	name	ingredients	diet	prep_time	cook_time	flavor_profile	course	state	region	No_of_ingredier
0	Balu shahi	Maida flour, yogurt, oil, sugar	vegetarian	45.0	25.0	sweet	dessert	West Bengal	East	
1	Boondi	Gram flour, ghee, sugar	vegetarian	80.0	30.0	sweet	dessert	Rajasthan	West	
2	Gajar ka halwa	Carrots, milk, sugar, ghee, cashews, raisins	vegetarian	15.0	60.0	sweet	dessert	Punjab	North	
3	Ghevar	Flour, ghee, kewra, milk, clarified butter, su...	vegetarian	15.0	30.0	sweet	dessert	Rajasthan	West	
4	Gulab jamun	Milk powder, plain flour, baking powder, ghee,...	vegetarian	15.0	40.0	sweet	dessert	West Bengal	East	
5	Imarti	Sugar syrup, lentil flour	vegetarian	10.0	50.0	sweet	dessert	West Bengal	East	
6	Jalebi	Maida, corn flour, baking soda, vinegar, curd,...	vegetarian	10.0	50.0	sweet	dessert	Uttar Pradesh	North	

	name	ingredients	diet	prep_time	cook_time	flavor_profile	course	state	region	No_of_ingredier
7	Kaju katli	Cashews, ghee, cardamom, sugar	vegetarian	10.0	20.0	sweet	dessert	NaN	NaN	
8	Kalakand	Milk, cottage cheese, sugar	vegetarian	20.0	30.0	sweet	dessert	West Bengal	East	
9	Kheer	Milk, rice, sugar, dried fruits	vegetarian	10.0	40.0	sweet	dessert	NaN	NaN	

## 8. Add a new column in the dataset to calculate the total time taken to make every dish.

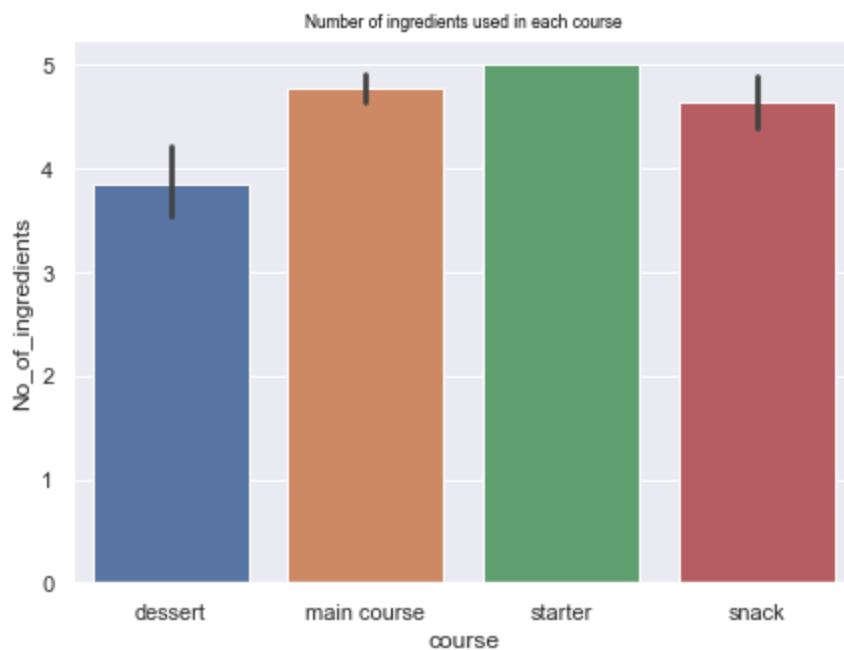
```
In [47]: df['totaltime'] = df['prep_time']+df['cook_time']
df.head()
```

```
Out[47]:
```

	name	ingredients	diet	prep_time	cook_time	flavor_profile	course	state	region	No_of_ingredients
0	Balu shahi	Maida flour, yogurt, oil, sugar	vegetarian	45.0	25.0	sweet	dessert	West Bengal	East	4
1	Boondi	Gram flour, ghee, sugar	vegetarian	80.0	30.0	sweet	dessert	Rajasthan	West	3
2	Gajar ka halwa	Carrots, milk, sugar, ghee, cashews, raisins	vegetarian	15.0	60.0	sweet	dessert	Punjab	North	6
3	Ghevar	Flour, ghee, kewra, milk, clarified butter, su...	vegetarian	15.0	30.0	sweet	dessert	Rajasthan	West	10
4	Gulab jamun	Milk powder, plain flour, baking powder, ghee,...	vegetarian	15.0	40.0	sweet	dessert	West Bengal	East	8

## 9. Barplot using seaborn library to display number of ingredients used in each course.

```
In [48]: plt.figure(figsize=(7,5))
sns.barplot(x=df['course'],y=df['No_of_ingredients']);
plt.title("Number of ingredients used in each course",color='black',fontsize='small');
```



We see that on an average starters require larger number of ingredients

## 10. Display the top 3 dishes with the most number of ingredients.

```
In [49]: df.nlargest(3, 'No_of_ingredients')
```

	name	ingredients	diet	prep_time	cook_time	flavor_profile	course	state	region	No_of_ingredients
3	Ghevar	Flour, ghee, kewra, milk, clarified butter, su...	vegetarian	15.0	30.0	sweet	dessert	Rajasthan	West	10
6	Jalebi	Maida, corn flour, baking soda, vinegar, curd,...	vegetarian	10.0	50.0	sweet	dessert	Uttar Pradesh	North	9
4	Gulab jamun	Milk powder, plain flour, baking powder, ghee,...	vegetarian	15.0	40.0	sweet	dessert	West Bengal	East	8

## 11. Give a total count of Ingredients that are most used in all the dishes and also visualize it by using barplot from seaborn library.

```
In [50]: print(df.ingredients.str.split(',').sum())
```

```
['Maida flour', ' yogurt', ' oil', ' sugar', 'Gram flour', ' ghee', ' sugar', 'Carrots', ' milk', ' sugar', ' ghee', ' cashews', ' raisins', 'Flour', ' ghee', ' kewra', ' milk', ' c
larified butter', ' sugar', ' almonds', ' pistachio', ' saffron', ' green cardamom', 'Milk
powder', ' plain flour', ' baking powder', ' ghee', ' milk', ' sugar', ' water', ' rose wa
ter', 'Sugar syrup', ' lentil flour', 'Maida', ' corn flour', ' baking soda', ' vinegar',
' curd', ' water', ' turmeric', ' saffron', ' cardamom', 'Cashews', ' ghee', ' cardamom',
' sugar', 'Milk', ' cottage cheese', ' sugar', 'Milk', ' rice', ' sugar', ' dried fruits',
'Gramflour', ' ghee', ' sugar', 'Yogurt', ' milk', ' nuts', ' sugar', 'Refined flour', '']
```

besan', ' ghee', ' powdered sugar', ' yoghurt', ' green cardamom', ' Firm white pumpkin',  
sugar', ' kitchen lime', ' alum powder', 'Rice', ' sugar', ' nuts', 'Condensed milk', ' su  
gar', ' spices', ' nuts', 'Semolina', ' ghee', ' nuts', ' milk', 'Khoas', ' coconut', ' mol  
u leaf', 'Corn flour', ' ghee', ' dry fruits', 'Gram flour', ' ghee', ' sugar', ' milk', '   
cardamom', 'Chhena', ' sugar', 'ghee', 'Chhena', ' sugar', ' milk', 'Sugar', ' chenna che  
ese', 'Flour', ' cream', ' sugar', ' saffron', ' lemon juice', ' coconut flakes', 'Chenn  
a', ' condensed milk', ' sugar', ' saffron', ' cardamom', 'Chhena', ' sugar', ' ghee', 'Fl  
our', ' fried milk power', ' sugar syrup', 'Yoghurt', ' refined flour', ' ghee', ' fennel  
seeds', 'Besan flour', ' sugar', ' ghee', 'Milk', ' jaggery', 'Chhena', ' sugar', ' ghee',  
' flour', 'Rice flour', ' wheat flour', 'Chenna', ' sweetened milk', 'Chhena', ' reduced m  
ilk', ' pistachio', 'Chhena', ' sugar', ' cardamom', 'Milk', ' sugar', ' saffron', ' carda  
mom', 'Rice flour', ' jaggery', ' ghee', ' vegetable oil', ' elachi', 'Rice flour', ' jagg  
ery', ' ghee', 'Besan', ' jaggery', ' cardamom powder', ' ghee', ' cashews and raisins', '   
jaggery syrup', ' sugar', 'Peanuts', ' jaggery', 'Milk', ' Sugar', ' Dharwadi buffalo mil  
k', 'Loaf bread', ' milk', 'Rice flour', ' sugar', ' salt', ' ghee', ' semolina', 'Wheat f  
lour', ' sugar', 'Black lentils', ' rice', 'Besan flour', ' semolina', ' mung bean', ' jag  
gery', ' coconut', ' skimmed milk powder', ' sugar', ' ghee', 'Maida flour', ' turmeric',  
' coconut', ' chickpeas', ' jaggery', ' ghee', ' cardamom', 'Rice flour', ' milk', 'Chana  
dal', ' jaggery', 'Rice', ' jaggery', ' cashews', ' ghee', 'Rice flour', ' powdered suga  
r', ' ghee', 'Apricots', ' sugar syrup', 'Vermicelli pudding', ' milk', 'Rice flour', ' ba  
nana', ' jaggery', ' coconut', 'Rice flour', ' jaggery', ' coconut', 'Rice flour', ' jagg  
ery', ' khus-khus seeds', 'Sugar', ' milk', ' nuts', 'Cucumber', ' rava', 'Milk', ' rice',  
' sugar', ' dry fruits', 'Semolina', ' sugar', 'Rice flour', ' coconut', ' jaggery', 'Sug  
ar', ' ghee', ' maida flour', ' semolina', 'Curd', ' sugar', ' saffron', ' cardamom', 'Mai  
da', ' sugar', ' ghee', 'Fish', ' potol', ' tomato', ' chillies', ' ginger', ' garlic', 'B  
oiled pork', ' onions', ' chillies', ' ginger and garlic', 'Rice', ' milk', ' sugar', ' ca  
rdamom', 'Rice', ' axone', ' salt', ' water', ' chillies', ' pork', 'Cauliflower', ' potat  
o', ' garam masala', ' turmeric', ' curry leaves', 'Rice flour', ' potato', ' bread crumb  
s', ' garam masala', ' salt', 'Potato', ' peas', ' chillies', ' ginger', ' garam masala',  
' garlic', 'Potato', ' fenugreek leaves', ' chillies', ' salt', ' oil', 'Potato', ' shimla  
mirch', ' garam masala', ' amchur powder', ' salt', 'Chole', ' rava', ' yogurt', ' plain f  
lour', ' baking soda', 'Ladies finger', ' garam masala', ' kasuri methi', ' tomatoes', ' c  
hili powder', 'Chicken thighs', ' basmati rice', ' star anise', ' sweet', ' green chillie  
s', 'Chicken', ' greek yogurt', ' cream', ' garam masala powder', ' cashew nuts', ' butte  
r', 'Chickpeas', ' tomato paste', ' garam masala', ' ginger', ' red onion', ' avocado oi  
l', 'Whole wheat flour', ' olive oil', ' hot water', ' all purpose flour', 'Chicken', ' da  
hi', ' sesame seeds', ' garam masala powder', ' cashew nuts', ' saffron', 'Naan bread', '   
tomato sauce', ' skinless chicken breasts', ' heavy cream', ' garam masala', 'Chicken', '   
whole wheat bread', ' rice flour', ' garam masala powder', ' whole egg', 'Chole', ' bhatu  
ra', ' garam masala', ' bay leaf', ' cinnamon stick', 'Moong dal', ' masoor dal', ' chana d  
al', ' wheat flour', ' almond', 'Moong dal', ' garam masala powder', ' garlic', ' green ch  
illi', ' all purpose flour', 'Red kidney beans', ' urad dal', ' cream', ' garam masala', '   
chili powder', 'Pigeon peas', ' garam masala', ' ginger', ' red onion', ' kasuri methi',  
'Baby potatoes', ' garam masala', ' cashew nuts', ' kasuri methi', ' tomatoes', 'Beaten ri  
ce flakes', ' potato', ' curry leaves', ' green chillies', ' lemon juice', 'Chana dal', ' w  
hole wheat flour', ' arhar dal', ' white urad dal', ' garam masala powder', 'Moong dal', '   
rava', ' garam masala', ' dough', ' fennel seeds', 'Cottage cheese', ' bell peppers', ' gr  
avy', ' garam masala', ' cashew nuts', 'Besan', ' garam masala powder', ' gram flour', ' g  
inger', ' curry leaves', 'Bitter gourd', ' fennel', ' garam masala powder', ' chili powde  
r', ' amchur powder', 'Moong dal', ' green peas', ' ginger', ' tomato', ' green chili', 'P  
aneer', ' potato', ' cream', ' corn flour', ' garam masala', 'Rose syrup', ' falooda sev',  
' mixed nuts', ' saffron', ' sugar', 'Bottle gourd', ' garam masala powder', ' gram flou  
r', ' ginger', ' chillies', 'Bottle gourd', ' coconut oil', ' garam masala', ' ginger', '   
green chillies', 'Wheat flour', ' roasted gram flour', ' tomato', ' nigella seeds', ' chil  
li', 'Palak', ' makki atta', ' mustard green', ' garam masala', ' ginger', 'Whole wheat fl  
our', ' chickpea flour', ' green chillies', 'Mushroom', ' malai', ' garam masala', ' ginge  
r', ' capsicum', 'Canned coconut milk', ' frozen green peas', ' wild mushrooms', ' garam m  
asala', ' tomatoes', 'Whole wheat flour', ' honey', ' butter', ' garlic', 'Green beans', '   
potatoes', ' khus khus', ' low fat', ' garam masala powder', 'Cottage cheese', ' palak', '   
cream', ' garam masala', ' butter', 'Paneer', ' whipping cream', ' garam masala', ' cashew  
nuts', ' butter', 'Paneer', ' greek yogurt', ' tandoori masala', ' cream', ' bell pepper',  
'Kala chana', ' mashed potato', ' boondi', ' sev', ' lemon', 'Whole wheat flour', ' musk m  
elon seeds', ' poppy seeds', ' edible gum', ' semolina', 'Urad dal', ' sev', ' lemon juic  
e', ' chopped tomatoes', 'Wheat flour', ' butter', ' potato', ' coriander', 'Arbi ke patt  
e', 'sesame seeds', ' gur', ' bengal gram flour', ' imli', 'Fennel', ' tea bags', ' tomat  
o', ' kasuri methi', ' cinnamon', 'Red kidney beans', ' garam masala powder', ' ginger', ' '

tomato', 'mustard oil', 'Garam masala powder', 'tomato', 'kasuri methi', 'cinnamon', 'mustard oil', 'Potatoes', 'green peas', 'garam masala', 'ginger', 'dough', 'Sattu', 'atta', 'dough', 'filling', 'mustard oil', 'Cottage cheese', 'malai', 'garam masala', 'ginger', 'tomato', 'Rose water', 'milk', 'white bread slices', 'saffron', 'almond s', 'Baby corn', 'french beans', 'garam masala', 'ginger', 'carrot', 'Greek yogurt', 'garam masala', 'kasuri methi', 'marinade', 'mustard oil', 'Chickpea flour', 'biryani masala powder', 'yogurt', 'fish fillets', 'green bell pepper', 'Whole wheat flour', 'arhar dal', 'ginger', 'kala jeera', 'green chilli', 'Raw banana', 'elephant foot yam', 'long beans', 'tindora', 'urad dal', 'Split pigeon peas', 'chana dal', 'urad dal', 'green peas', 'french beans', 'Chana dal', 'urad dal', 'fresh coconut', 'sesame seeds', 'curry leaves', 'Chana dal', 'urad dal', 'whole urad dal', 'blend rice', 'rock salt', 'Rice flour', 'hot water', 'grated coconut', 'Split urad dal', 'urad dal', 'idli rice', 'thick poha', 'rock salt', 'Carrot', 'yellow mustard', 'red chilli', 'black salt', 'Sesame oil', 'drumstick', 'tamarind paste', 'sambar powder', 'tomato', 'Moong dal', 'chana dal', 'spinach', 'urad dal', 'coconut oil', 'Urad dal', 'curry leaves', 'sugar', 'mustard seeds', 'spinach', 'Greens', 'tomato', 'mustard seeds', 'fenugreek seeds', 'Amaranth leaves', 'split urad dal', 'mustard seeds', 'grated coconut', 'red chilli', 'Beef', 'coconut', 'garam masala', 'curry leaves', 'green chilies', 'chili powder', 'Chana dal', 'urad dal', 'potato', 'beans', 'peas', 'Moong dal', 'chana dal', 'cabbage', 'tamarind', 'curry leaves', 'Moong dal', 'cucumber', 'curry leaves', 'green chili', 'lemon juice', 'Chana dal', 'urad dal', 'gooseberry', 'raw rice', 'curry leaves', 'Sesame oil', 'raw rice', 'jaggery', 'grated coconut', 'Pearl onions', 'urad dal', 'drumsticks', 'tomato', 'curry leaves', 'Chana dal', 'urad dal', 'potatoes', 'idli rice', 'thick poha', 'Coconut oil', 'cucumber', 'curd', 'curry leaves', 'mustard seeds', 'Yogurt', 'ginger', 'curry leaves', 'baking soda', 'green chilli', 'Lentils', 'black pepper', 'vegetable oil', 'Raw rice', 'jaggery', 'milk', 'Rice', 'cashew nuts', 'milk', 'raisins', 'sugar', 'Arhar dal', 'sambar powder', 'tomato', 'curry leaves', 'fennel seeds', 'Green moong beans', 'rice flour', 'Chana dal', 'urad dal', 'beans', 'coconut', 'mustard', 'Urad dal', 'lemon', 'tamarind', 'cooked rice', 'curry leaves', 'Tomato', 'curry leaves', 'garlic', 'mustard seeds', 'hot water', 'Brown rice flour', 'sugar', 'grated coconut', 'Pigeon peas', 'eggplant', 'drumsticks', 'sambar powder', 'tamarind', 'Thin rice flakes', 'black sesame seeds', 'curry leaves', 'Sevai', 'parboiled rice', 'steamer', 'Urad dal', 'curd', 'sesame oil', 'ginger', 'curry leaves', 'mustard seeds', 'Coconut', 'whole red beans', 'masala', 'sesame oil', 'tamarind', 'Chana dal', 'urad dal', 'thick poha', 'tomato', 'butter', 'Urad dal', 'ginger', 'curry leaves', 'green chilies', 'black pepper', 'Meat curry powder', 'chicken chunks', 'ginger', 'tomato', 'cinnamon', 'Chana dal', 'urad dal', 'ginger', 'curry leaves', 'sugar', 'Kala masala', 'arhar dal', 'curry leaves', 'mustard seeds', 'hot water', 'Gram flour', 'mustard', 'garlic', 'turmeric', 'red chilli', 'Baingan', 'fish', 'coconut oil', 'fresh coconut', 'ginger', 'Urad dal', 'potatoes', 'wheat flour', 'sooji', 'Wheat flour', 'pearl millet flour', 'hot water', 'Condensed milk', 'mawa', 'desiccated coconut', 'almonds', 'cashews', 'Jowar flour', 'sesame seeds', 'Bombay duck', 'malvani masala', 'rice flour', 'bombay rava', 'green chilies', 'Rice flour', 'sesame', 'plain flour', 'turmeric', 'red chilli', 'Citric acid', 'fry', 'raisins', 'sugar', 'chana dal', 'Urad dal', 'bengal gram flour', 'dried mango', 'baking soda', 'black salt', 'Condensed milk', 'nestle cream', 'coconut ice', 'red food coloring', 'desiccated coconut', 'Whole wheat flour', 'dal', 'kokum', 'gur', 'bengal gram flour', 'Pav', 'aloo', 'peanut', 'pomegranate', 'star anise', 'Urad dal', 'bhuna chana', 'garam masala', 'dates', 'tamarind', 'Arhar dal', 'coconut oil', 'curry leaves', 'mustard seeds', 'red chilli', 'Rava', 'coconut', 'gram flour', 'mustard', 'sesame', 'Bottle gourd', 'green', 'raisins', 'sugar', 'clarified butter', 'Yogurt', 'besan', 'sauce', 'garam masala powder', 'gram flour', 'Wheat flour', 'jaggery', 'clarified butter', 'sliced almonds', 'Dry fruits', 'semolina', 'all purpose flour', 'Bottle gourd', 'chana dal', 'cabbage', 'urad dal', 'toor dal', 'Whole wheat rava', 'chia seed', 'lemon', 'edible gum', 'litre milk', 'Green chilies', 'lemon juice', 'chili powder', 'boiled potatoes', 'Wheat flour', 'cashews', 'rapeseed oil', 'Mango', 'sugar', 'Whole wheat flour', 'low fat', 'bengal gram flour', 'Green chili paste', 'white sesame seeds', 'gram flour', 'curry leaves', 'green chili', 'Rice flour', 'urad dal', 'wheat flour', 'gram flour', 'turmeric', 'Cinnamon', 'jaggery', 'clarified butter', 'dry roasted', 'Cucumber', 'carrot', 'tomatoes', 'cilantro', 'Rava', 'gram flour', 'lemon juice', 'turmeric', 'fenugreek leaves', 'Rose water', 'pistachio', 'badam', 'bengal gram flour', 'saffron', 'Bottle gourd', 'whole wheat flour', 'rava', 'sesame seeds', 'bengal gram flour', 'Arbi ke patte', 'sesame seeds', 'gur', 'bengal gram flour', 'imli', 'Pav bhaji masala', 'gobi', 'potatoes', 'green peas', 'dinner rolls', 'Aloo', 'urad dal', 'mustard', 'ginger', 'curry leaves', 'Raw peanuts', 'sabudana', 'lemon', 'avocado oil', 'curry leaves', 'green chili', 'Khaman', 'pomegranate', 'sev', 'powdered sugar', 'garlic', 'Sev', 'ginger',



'tomato', 'sugar', 'Wheat flour', 'baking soda', 'all purpose flour', 'black pepper', 'sunflower oil', 'Whole wheat flour', 'gur', 'clarified butter', 'Rice flakes', 'yogurt', 'raw rice', 'jaggery', 'grated coconut', 'Whole wheat flour', 'rice flour', 'pearl millet flour', 'sorghum flour', 'sesame seeds', 'Sweet potato', 'surti papdi', 'baby potatoes', 'valor papdi', 'green peas', 'Gobi', 'potato', 'beans', 'khus khus', 'coconut', 'Chicken', 'coconut oil', 'wine vinegar', 'ginger', 'green', 'cinnamon', 'Green garlic chutney', 'fresh green peas', 'ginger', 'lemon juice', 'plain flour', 'Moong beans', 'jaggery', 'red chillies', 'oil', 'salt', 'Rice flour', 'sesame seeds', 'baking soda', 'peanut oil', 'Chickpea flour', 'methi leaves', 'jowar flour', 'wheat flour', 'Semolina', 'clarified butter', 'oil', 'white flour', 'black pepper', 'Yogurt', 'fresh coconut', 'sesame seeds', 'semolina', 'gram flour', 'Ridge gourd', 'baking soda', 'sugar', 'grated coconut', 'peas', 'Whole wheat flour', 'khus khus', 'sesame seeds', 'dry coconut', 'gur', 'Rice', 'mango', 'curd', 'Sticky rice', 'rice flour', 'jaggery', 'orange rind', 'Raw papaya', 'panch phoran masala', 'nigella seeds', 'mustard oil', 'fennel seeds', 'Rice', 'eggs', 'carrot', 'beetroot', 'Maida', 'vegetable oil', 'Potatoes', 'mustard oil', 'fish', 'green chillies', 'Ridge gourd', 'fish', 'lemon', 'tomatoes', 'mustard oil', 'Brinjal', 'onions', 'salt', 'sesame seeds', 'coriander', 'Potatoes', 'garam masala', 'tomatoes', 'mustard oil', 'bay leaf', 'Forbidden black rice', 'chicken', 'olive oil', 'slivered almonds', 'garlic powder', 'Biryani masala', 'mixed vegetables', 'yellow moong daal', 'whole red', 'mustard seeds', 'Brown rice', 'soy sauce', 'olive oil', 'Coconut milk', 'lobster', 'fresh green chilli', 'ginger', 'red onion', 'Baking soda', 'clarified butter', 'oil', 'all purpose flour', 'Jaggery', 'raisins', 'Lamb', 'garam masala powder', 'curd', 'turmeric', 'bay leaf', 'Coconut', 'prawns', 'curd', 'mustard seed', 'green chili', 'Fish fillet', 'besan', 'lemon', 'mint', 'ginger', 'Fermented bamboo shoot', 'potato', 'ginger', 'green', 'mustard oil', 'Banana flower', 'chicken', 'green chili', 'mustard oil', 'lemon juice', 'Aloo', 'tomatoes', 'mustard oil', 'bay leaf', 'cinnamon stick', 'Rice flour', 'mutton', 'banana', 'gram flour', 'olive oil', 'baking powder', 'Fish roe', 'pumpkin flowers', 'mustard oil', 'turmeric', 'tomato', 'Chana dal', 'fresh coconut', 'ginger', 'cinnamon', 'raisins', 'Curd', 'cooked rice', 'curry leaves', 'dry chilli', 'Tea leaves', 'white sesame seeds', 'dry coconut', 'soaked rice', 'Basmati rice', 'rose water', 'sugar', 'clarified butter', 'cardamom pods', 'Coconut milk', 'prawns', 'garlic', 'turmeric', 'sugar', 'Red pepper', 'red onion', 'butter', 'watercress', 'olive oil', 'Green beans', 'bitter gourd', 'ridge gourd', 'banana', 'brinjal', 'Glutinous rice', 'black sesame seeds', 'gur', 'Coconut milk', 'egg yolks', 'clarified butter', 'all purpose flour', 'Cottage cheese', 'dry dates', 'dried rose petals', 'pistachio', 'badam', 'Milk powder', 'dry fruits', 'arrowroot powder', 'all purpose flour', 'Brown rice', 'fennel seeds', 'grated coconut', 'black pepper', 'ginger powder']

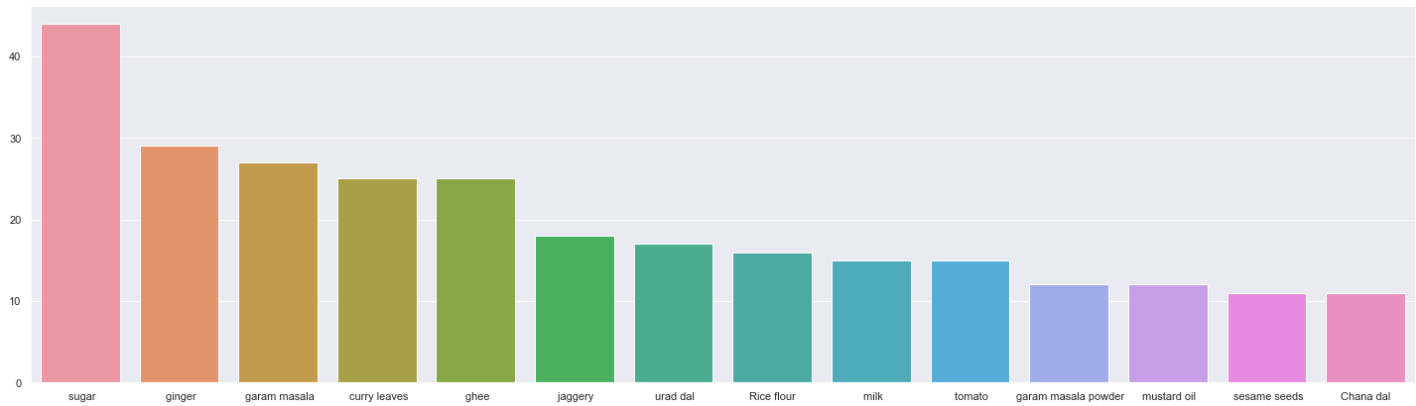
```
In [51]: # Ingredients that are most used in all the dishes
ingredients = pd.Series(df.ingredients.str.split(',').sum()).value_counts()
ingredients = ingredients[ingredients>10]
ingredients
```

```
Out[51]: sugar          44
ginger          29
garam masala    27
curry leaves    25
ghee            25
jaggery         18
urad dal        17
Rice flour      16
milk            15
tomato          15
garam masala powder  12
mustard oil     12
sesame seeds    11
Chana dal       11
dtype: int64
```

```
In [52]: f, ax = plt.subplots(figsize=(25,7))
sns.barplot(ingredients.index, ingredients.values)
```

Pass the following variables as keyword args: x, y. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.

Out[52]: <AxesSubplot:>



We observe that sugar is most used in all the dishes.

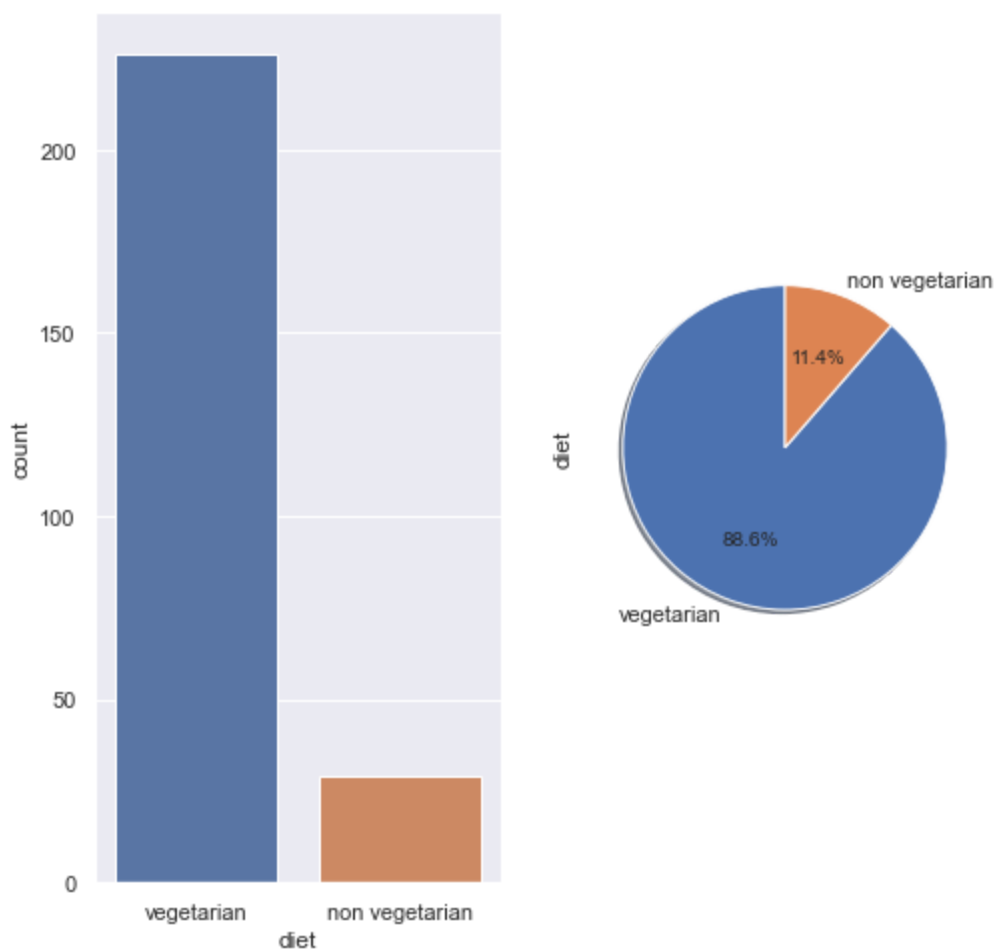
## 12. Visualize the number of dishes that are vegetarians and the non-vegetarians and find out the percentage of veg and non veg dishes using pie chart. Write the inference about it.

```
In [53]: fig1, ax1 = plt.subplots(1,2,figsize=(8,8))
sns.countplot(df['diet'],ax=ax1[0])
labels = 'vegetarian', 'non vegetarian'
df.diet.value_counts().plot.pie(labels=labels, autopct='%1.1f%%',shadow=True, startangle=0)
```

C:\Users\BJCRC\anaconda3\lib\site-packages\seaborn\\_decorators.py:36: FutureWarning:

Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.

Out[53]: <AxesSubplot:ylabel='diet'>



In the data, vegetarian food is 88.6%. Most of Indian would like to eat vegetarian food.

### 13. Repeat Q.12 for all categorical features with inferences.

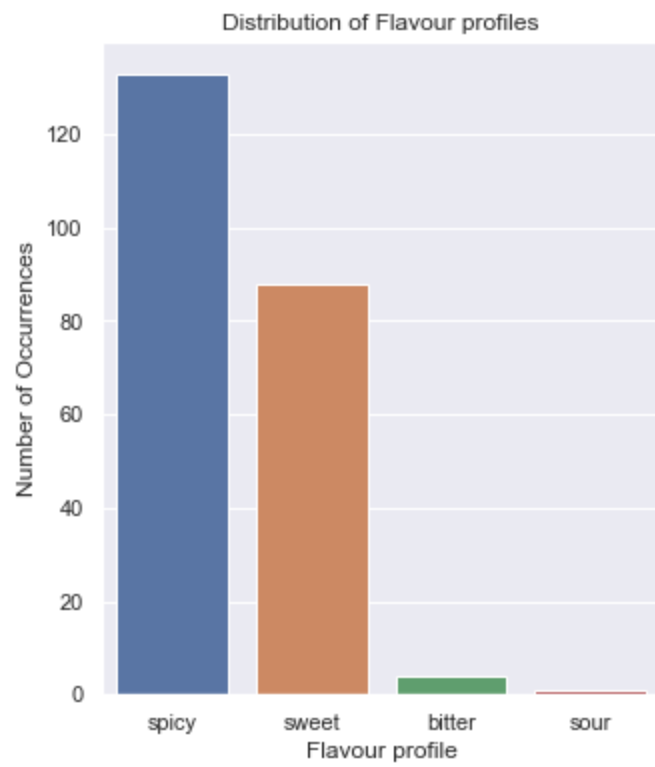
Flavour Profile

In [54]:

```
carrier_count = df['flavor_profile'].value_counts()
f, ax = plt.subplots(figsize=(5,6))
sns.barplot(carrier_count.index, carrier_count.values)
plt.title('Distribution of Flavour profiles')
plt.ylabel('Number of Occurrences', fontsize=12)
plt.xlabel('Flavour profile', fontsize=12)
plt.show()
pie_df = df.flavor_profile.value_counts().reset_index()
pie_df.columns = ['flavor_profile', 'count']
fig = px.pie(pie_df, values='count', names='flavor_profile', title='pie chart of different
fig.show()
```

C:\Users\BJCRC\anaconda3\lib\site-packages\seaborn\\_decorators.py:36: FutureWarning:

Pass the following variables as keyword args: x, y. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.



Inference: We see that most dishes in Indian cuisine, i.e, about 59% of them are spicy ; 39% of them are sweet ; 2% of them are bitter while only about 1% of the dishes are sour.

Courses

```
In [55]: carrier_count = df['course'].value_counts()
```

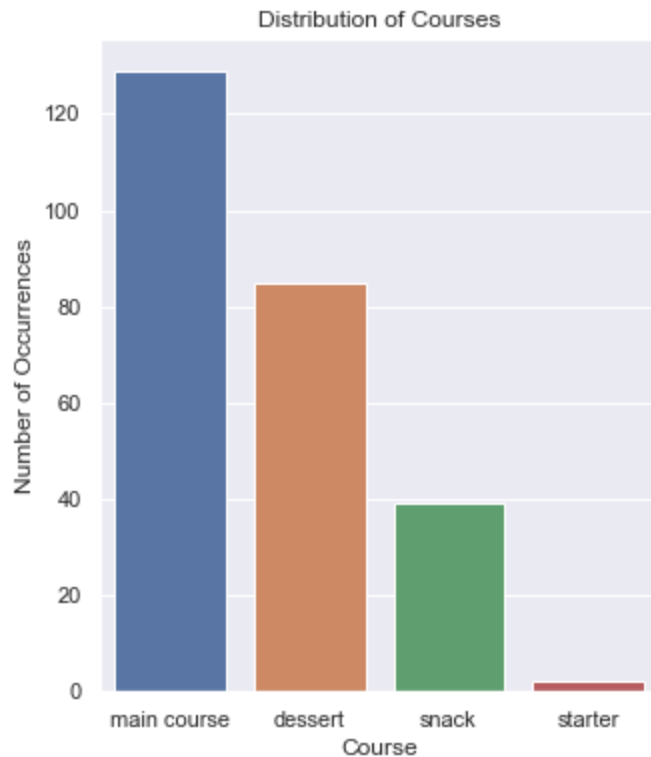
```

f, ax = plt.subplots(figsize=(5,6))
sns.barplot(carrier_count.index, carrier_count.values)
plt.title('Distribution of Courses')
plt.ylabel('Number of Occurrences', fontsize=12)
plt.xlabel('Course', fontsize=12)
plt.show()
pie_df = df.course.value_counts().reset_index()
pie_df.columns = ['course', 'count']
fig = px.pie(pie_df, values='count', names='course', title='pie chart of different courses')
fig.show()

```

C:\Users\BJCRC\anaconda3\lib\site-packages\seaborn\\_decorators.py:36: FutureWarning:

Pass the following variables as keyword args: x, y. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.



Inference: We see that most dishes in Indian cuisine, i.e, about 51% of them are main course dishes; 34% of them are deserts ; 16%% of them are snacks while only about 1% of the dishes are starters.

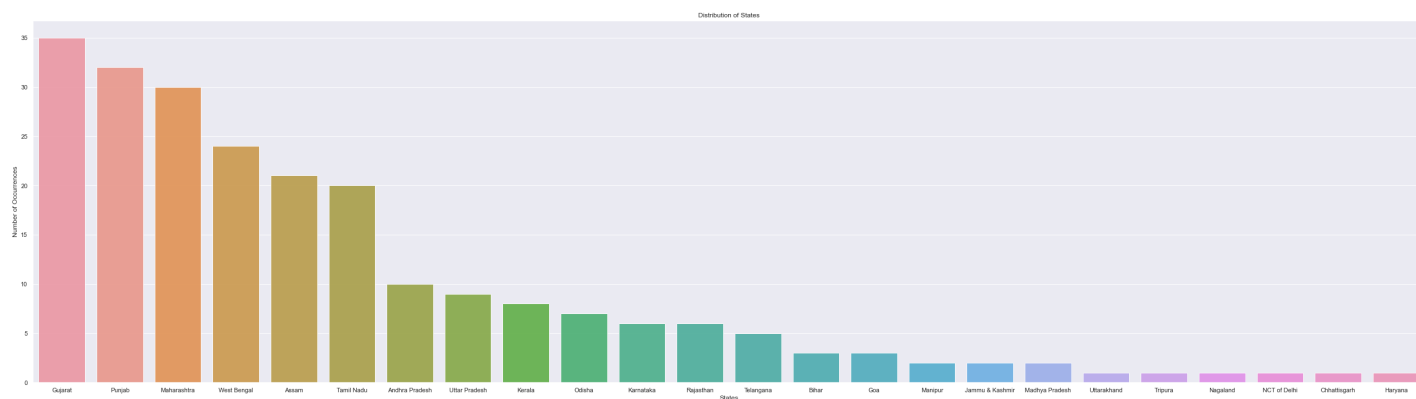
## State

In [56]:

```
carrier_count = df['state'].value_counts()
sns.set(style="darkgrid")
f, ax = plt.subplots(figsize=(45,12))
sns.barplot(carrier_count.index, carrier_count.values, alpha=0.9)
plt.title('Distribution of States')
plt.ylabel('Number of Occurrences', fontsize=12)
plt.xlabel('States', fontsize=12)
plt.show()
pie_df = df.state.value_counts().reset_index()
pie_df.columns = ['state', 'count']
fig = px.pie(pie_df, values='count', names='state', title='Pie chart of different states')
fig.show()
```

C:\Users\BJCRC\anaconda3\lib\site-packages\seaborn\\_decorators.py:36: FutureWarning:

Pass the following variables as keyword args: x, y. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.



Inference: We see that most dishes in Indian cuisine data set , i.e, about 15% of them are from Gujarat.

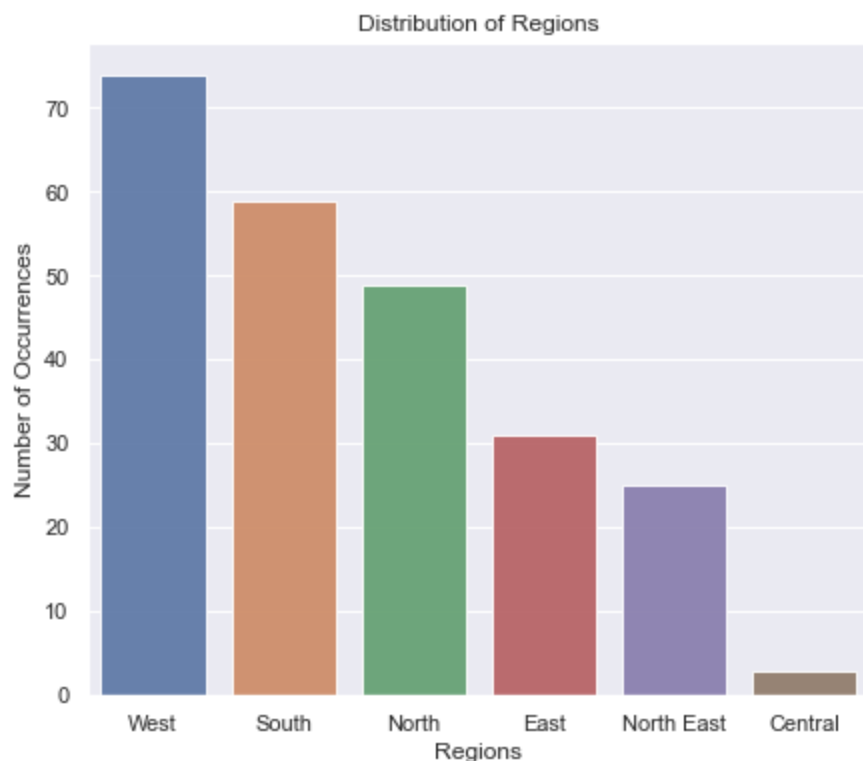
Region

In [57]:

```
carrier_count = df['region'].value_counts()
sns.set(style="darkgrid")
f, ax = plt.subplots(figsize=(7,6))
sns.barplot(carrier_count.index, carrier_count.values, alpha=0.9)
plt.title('Distribution of Regions')
plt.ylabel('Number of Occurrences', fontsize=12)
plt.xlabel('Regions', fontsize=12)
plt.show()
pie_df = df.region.value_counts().reset_index()
pie_df.columns = ['region', 'count']
fig = px.pie(pie_df, values='count', names='region', title='pie chart of different regions')
fig.show()
```

C:\Users\BJCRC\anaconda3\lib\site-packages\seaborn\\_decorators.py:36: FutureWarning:

Pass the following variables as keyword args: x, y. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.



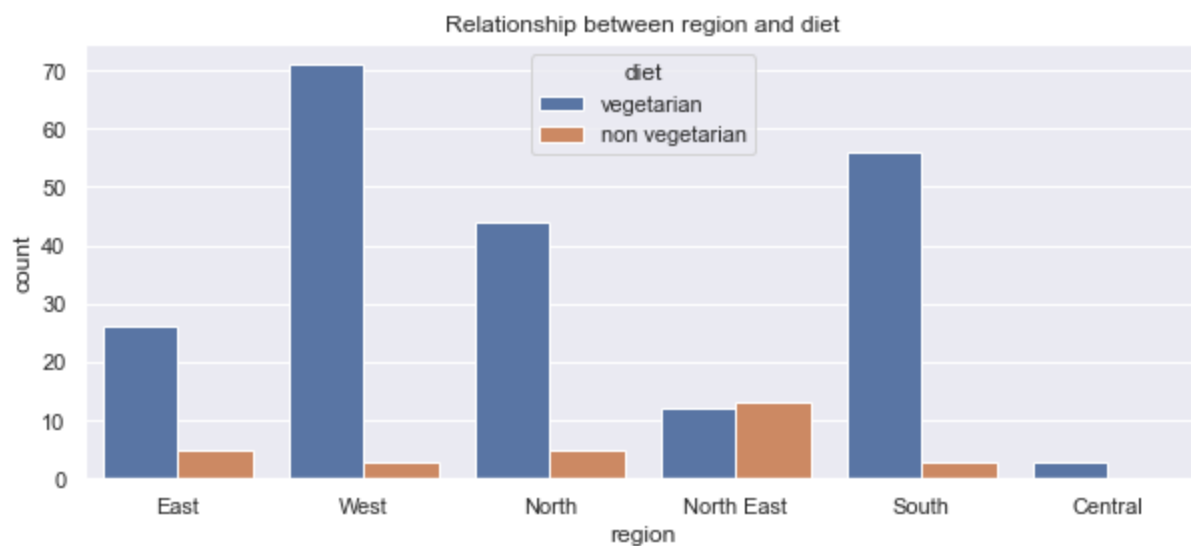
Inference: We see that most dishes in Indian cuisine data set , i.e, about 31% of them are from Western region of India .

## 14. Analyze the relationship between all categorical features with inferences.

Analysis for all the different regions

In [58]:

```
#Relationship between region and diet
fig, axes = plt.subplots(figsize=(10, 4))
sns.countplot(x='region', hue='diet', data=df)
axes.set_title('Relationship between region and diet')
plt.show()
```





Inference: We see that :

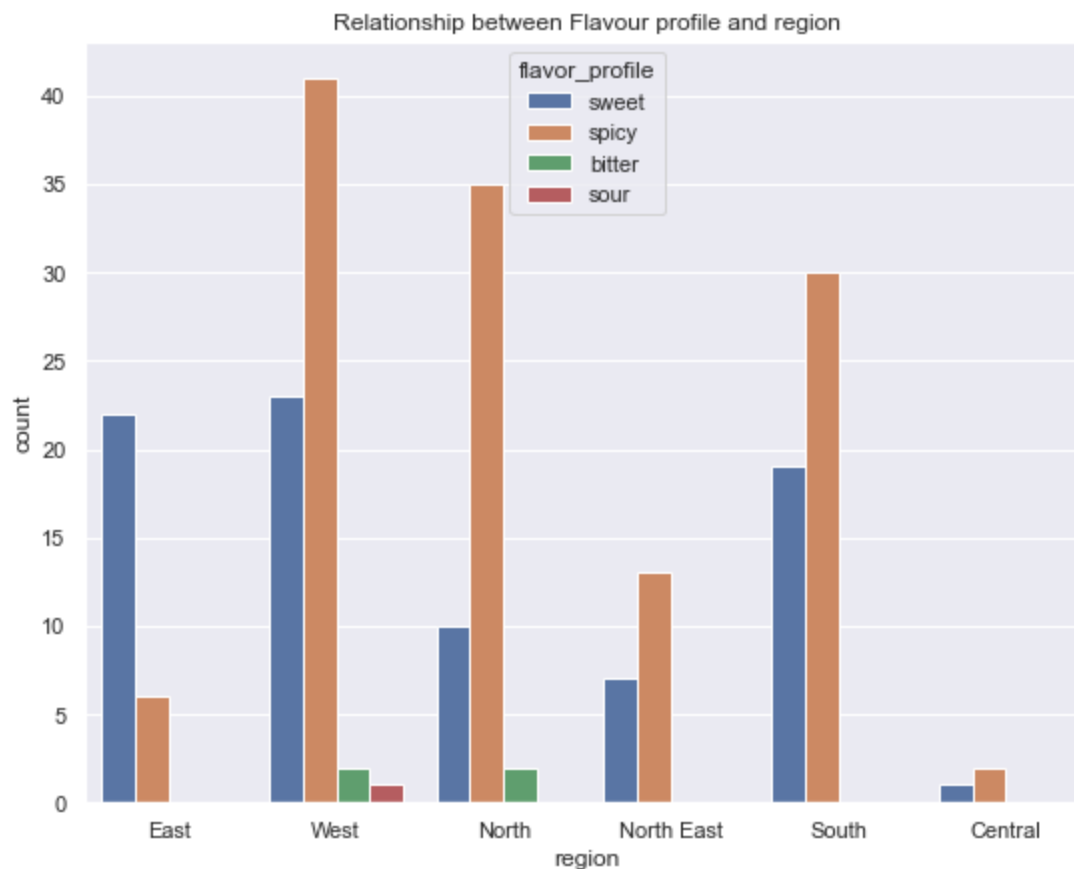
1)More of vegetarian food is eaten in the western and southern parts of India

2)More of non vegetarian food is eaten in the North East parts of India

3) In the Central region only Vegetarian food is eaten

In [59]:

```
#Relationship between flavour profile and region
fig, axes = plt.subplots(figsize=(9,7))
sns.countplot(x='region', hue='flavor_profile', data=df)
axes.set_title('Relationship between Flavour profile and region')
plt.show()
```



Inference:

1)In most of the regions spicy flavour is the most preferred.

2) Most number of spicy and sweet dishes come from the western region.

3)In the Eastern regions sweet flavour is more preferred.

4)Sour dishes are found to come from the western region only.

Analysis for all different states

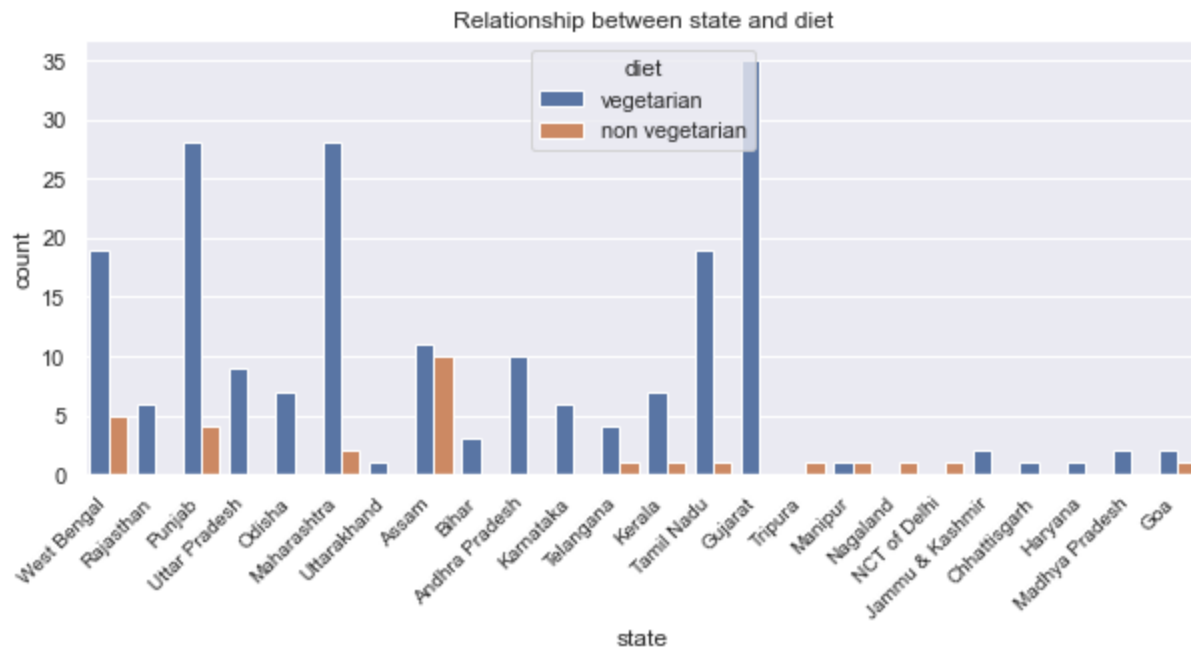
In [60]:

```
#Relationship between state and diet
fig, axes = plt.subplots(figsize=(10, 4))
sns.countplot(x='state', hue='diet', data=df)
axes.set_title('Relationship between state and diet')
plt.xticks(
    rotation=45,
    horizontalalignment='right',
    fontweight='light',
```

```

        fontsize='small'
    )
    plt.show()

```



Inference:

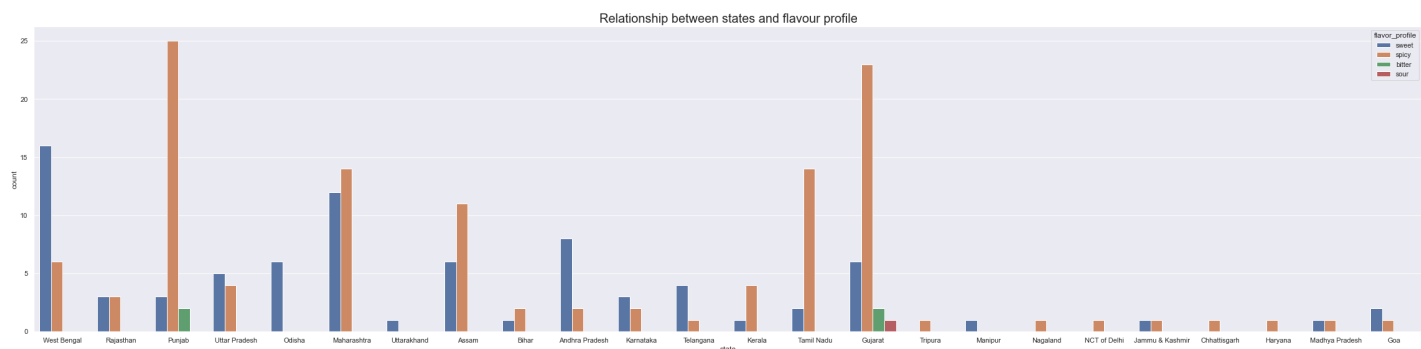
- 1) States like Gujarat, Andhra Pradesh, Karnataka, Uttar Pradesh, Odisha, J&K, Haryana, Chhattisgarh, Madhya Pradesh, Rajasthan and Bihar have only vegetarian food.
- 2) Most number of vegetarian dishes come from Gujarat
- 3) States like Tripura, Delhi and Nagaland have only non vegetarian food.
- 4) Most number of non vegetarian dishes come from Assam
- 5) Others have both non veg and veg food.

In [61]:

```

#Relationship between states and flavour profile
fig, axes = plt.subplots(figsize=(40,9))
sns.countplot(x='state', hue='flavor_profile', data=df)
axes.set_title('Relationship between states and flavour profile', fontsize=20)
plt.show()

```



Inference:

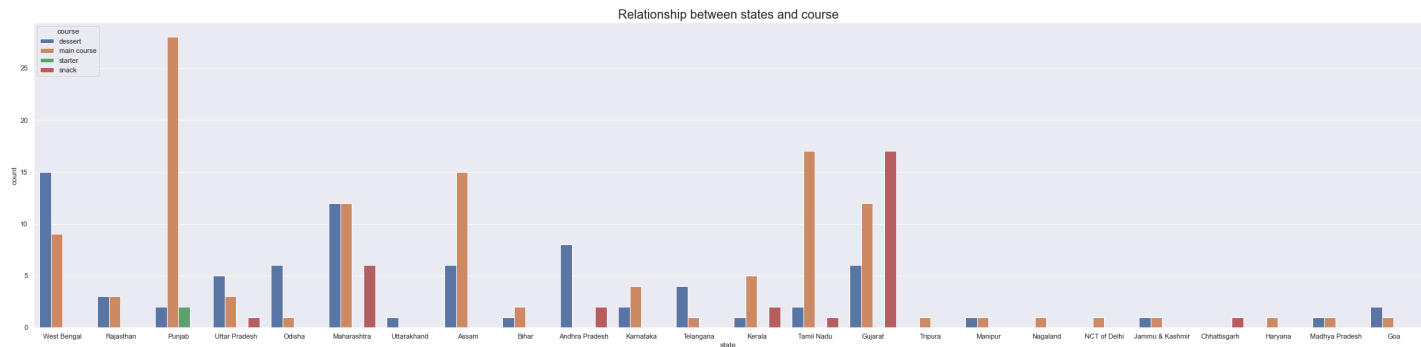
- 1) Most number of spicy dishes come from Punjab.
- 2) Most number of sweet dishes come from West Bengal.

3) More number of bitter dishes come from Guajrat

4) Sour dishes are only originating from Guajrat

In [62]:

```
#Relationship between states and courses
fig, axes = plt.subplots(figsize=(40,9))
sns.countplot(x='state', hue='course', data=df)
axes.set_title('Relationship between states and course', fontsize=20)
plt.show()
```



Inference:

1) Most main course dishes come from Punjab

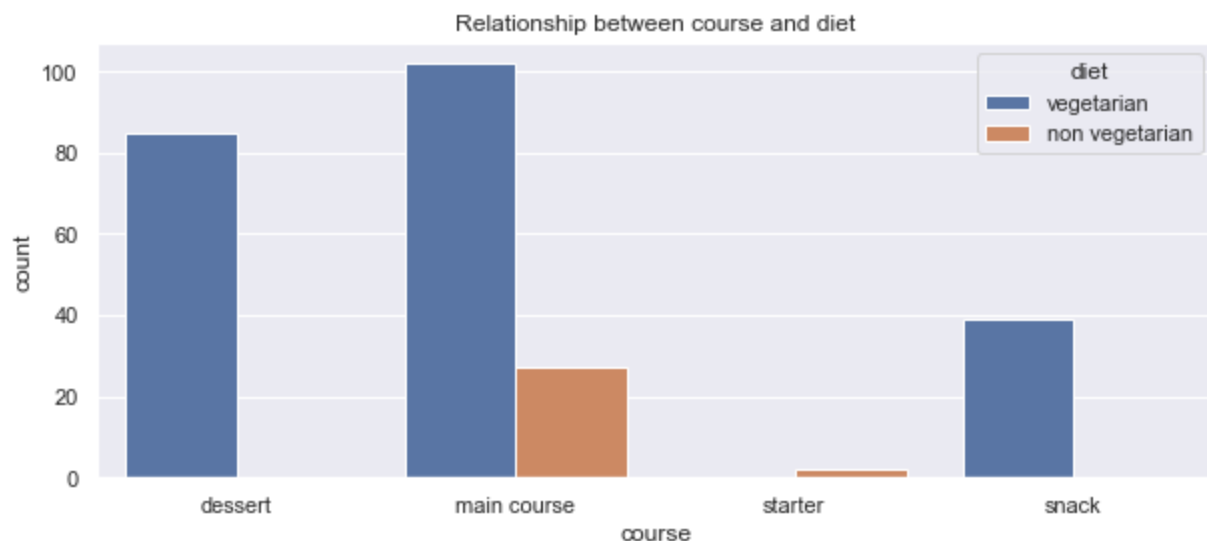
2) Starters are also only from punjab

3) Deserts from West Baengal are more in number

4) Most number of snacks also come from Gujrat

In [63]:

```
#Relationship between course and diet
fig, axes = plt.subplots(figsize=(10, 4))
sns.countplot(x='course', hue='diet', data=df)
axes.set_title('Relationship between course and diet')
plt.show()
```



Inference: We see that:

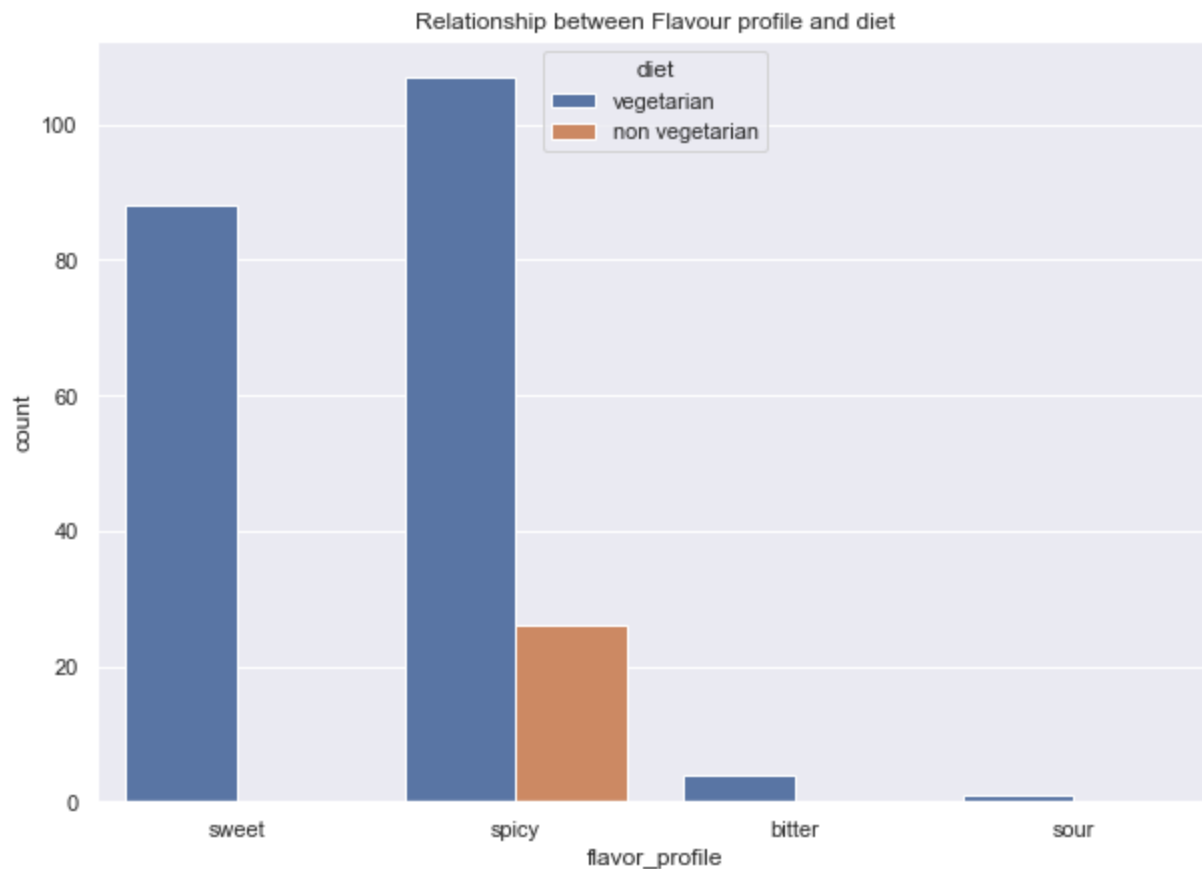
1) Deserts and snacks consist of only vegetarian food

2) Main course consists mostly of vegetarian food

### 3) Starters consist of only non vegetarian food

In [64]:

```
#Relationship between flavour profile and diet
fig, axes = plt.subplots(figsize=(10,7))
sns.countplot(x='flavor_profile', hue='diet', data=df)
axes.set_title('Relationship between Flavour profile and diet')
plt.show()
```

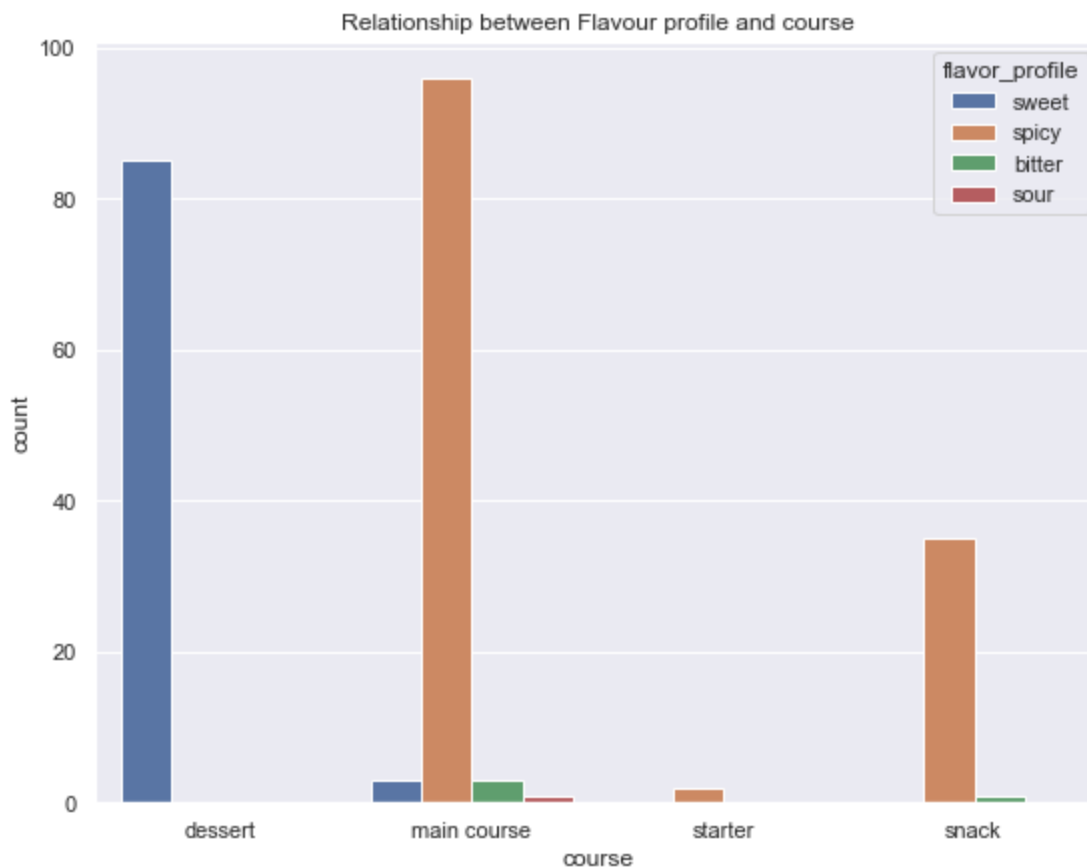


Inference:

- 1) Dishes of sweet, sour and bitter flavours are completely vegetarian
- 2) There are more vegetarian spicy dishes than there are non vegetarian

In [65]:

```
#Relationship between flavour profile and course
fig, axes = plt.subplots(figsize=(9,7))
sns.countplot(x='course', hue='flavor_profile', data=df)
axes.set_title('Relationship between Flavour profile and course')
plt.show()
```



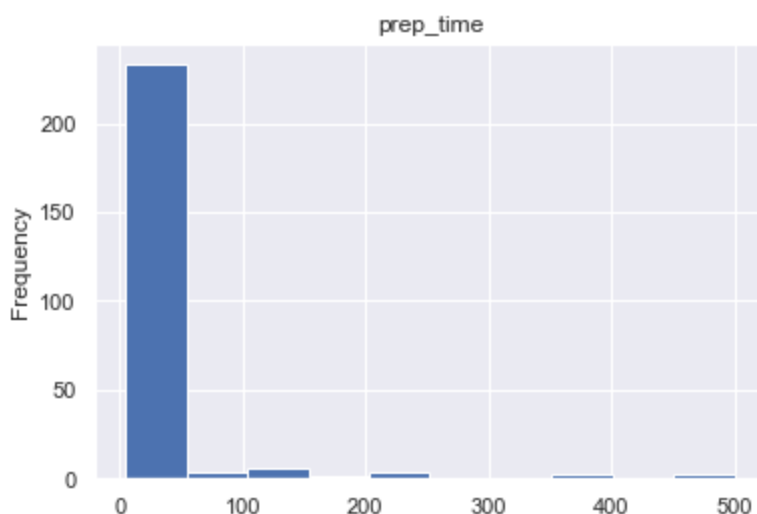
Inference:

- 1) Most of the main course dishes and snacks are spicy
- 2) The starters consist of only spicy dishes
- 3) Bitter and sour dishes are found in the maincourse only

## 15. Analysis of preparation time , cooking time and total time- Plot a histogram for preparation time the number of dishes taking different ranges of the preparation time

```
In [66]: fig, axes = plt.subplots()
axes.set_title('prep_time')
df.prep_time.plot(kind='hist')
```

```
Out[66]: <AxesSubplot:title={'center':'prep_time'}, ylabel='Frequency'>
```

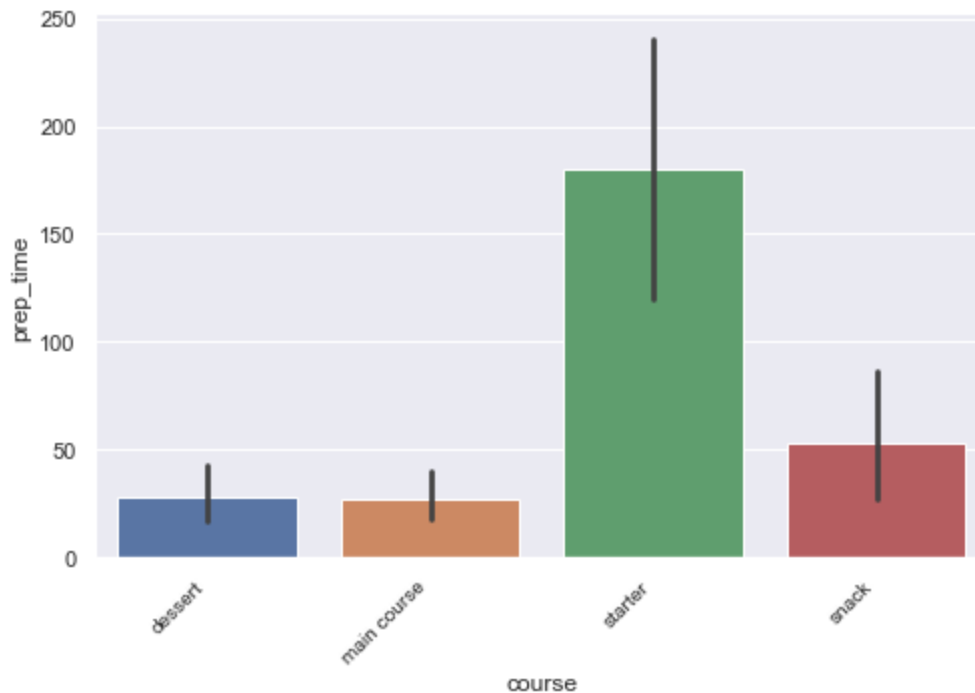


Inference: We can say that most dishes require a preparation time between 10 to 50 mins.

## 16. Analyze the preparation time and cooking time required for each course.

```
In [67]: ig=plt.figure(figsize=(8,5))
sns.barplot(x='course',y='prep_time',data=df)
plt.xticks(
    rotation=45,
    horizontalalignment='right',
    fontweight='light',
    fontsize='small'
)
```

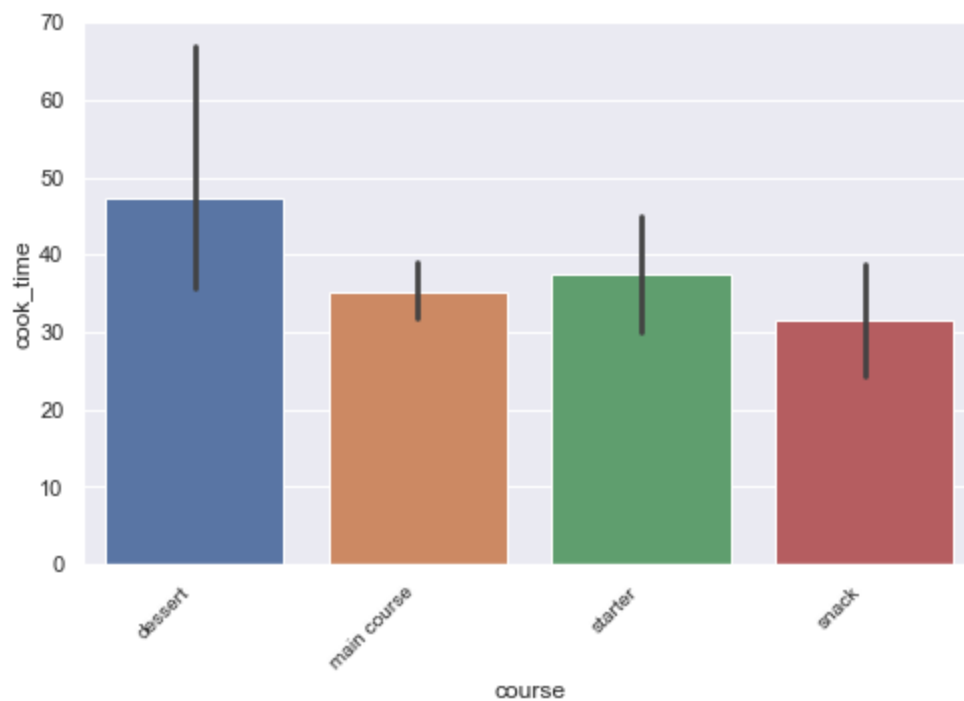
```
Out[67]: (array([0, 1, 2, 3]),
 [Text(0, 0, 'dessert'),
  Text(1, 0, 'main course'),
  Text(2, 0, 'starter'),
  Text(3, 0, 'snack')])
```



Inference: We can say that starter requires most time in preparation

```
In [68]: ig=plt.figure(figsize=(8,5))
sns.barplot(x='course',y='cook_time',data=df)
plt.xticks(
    rotation=45,
    horizontalalignment='right',
    fontweight='light',
    fontsize='small'
)
```

```
Out[68]: (array([0, 1, 2, 3]),
 [Text(0, 0, 'dessert'),
  Text(1, 0, 'main course'),
  Text(2, 0, 'starter'),
  Text(3, 0, 'snack')])
```

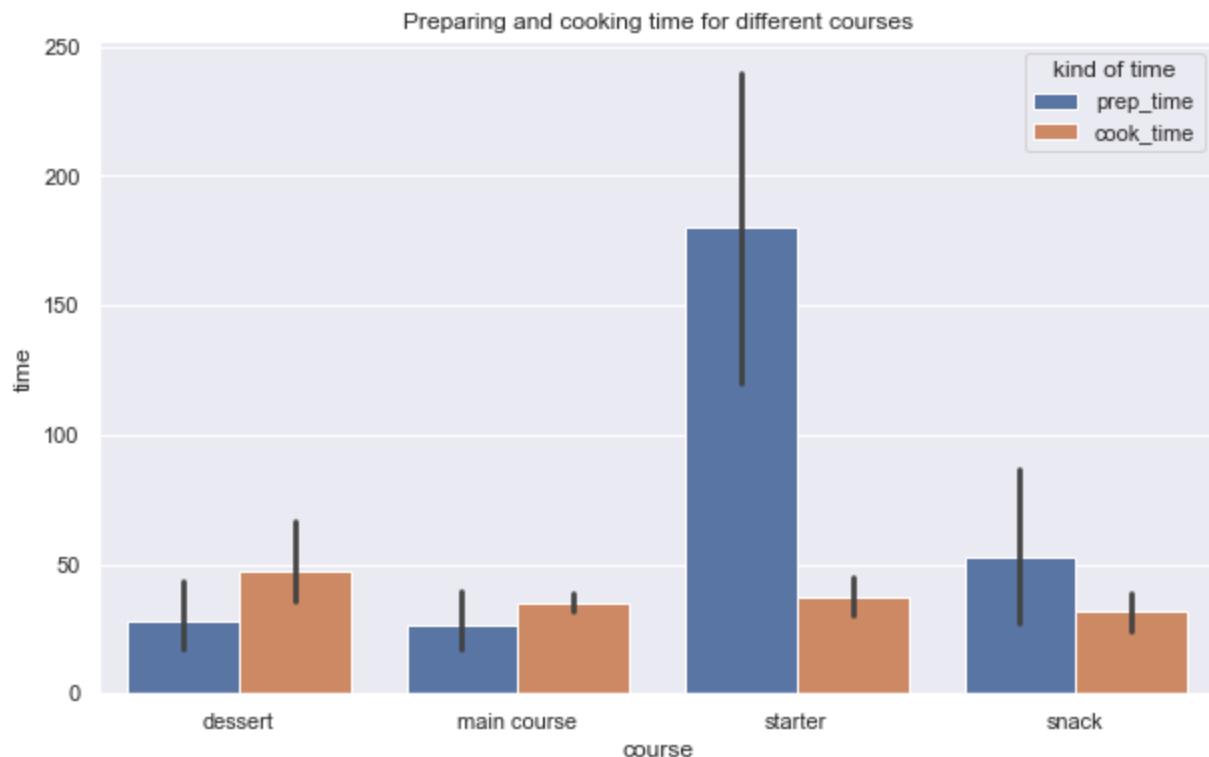


Inference: We Observe that dessert take maximum cooking time

## 17. Compare the preparation time and cook time for each course.

In [69]:

```
df_time = df[['prep_time', 'cook_time', 'course']]
df_time_melt = pd.melt(df_time, id_vars='course', value_vars=['prep_time', 'cook_time'],
                        df_time_melt
fig, axes = plt.subplots(figsize = (10, 6))
sns.barplot( x='course', y='time', hue = 'kind of time', data = df_time_melt)
axes.set_title('Preparing and cooking time for different courses')
plt.show()
```



Inference :

1) It takes longer to prepare for starters than any other course.

2) It takes much lesser time to cook the starters and snacks than to prepare them.

3) While it takes longer to cook desserts and main course dishes than it takes to prepare for them

## 18. Compare cooking time and preparation time for dishes of certain course and certain flavor profile.

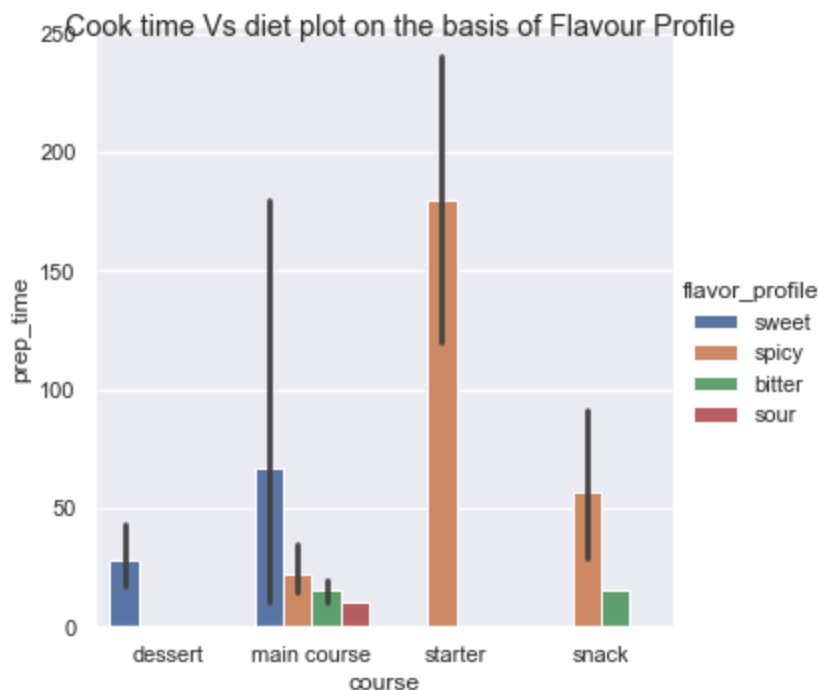
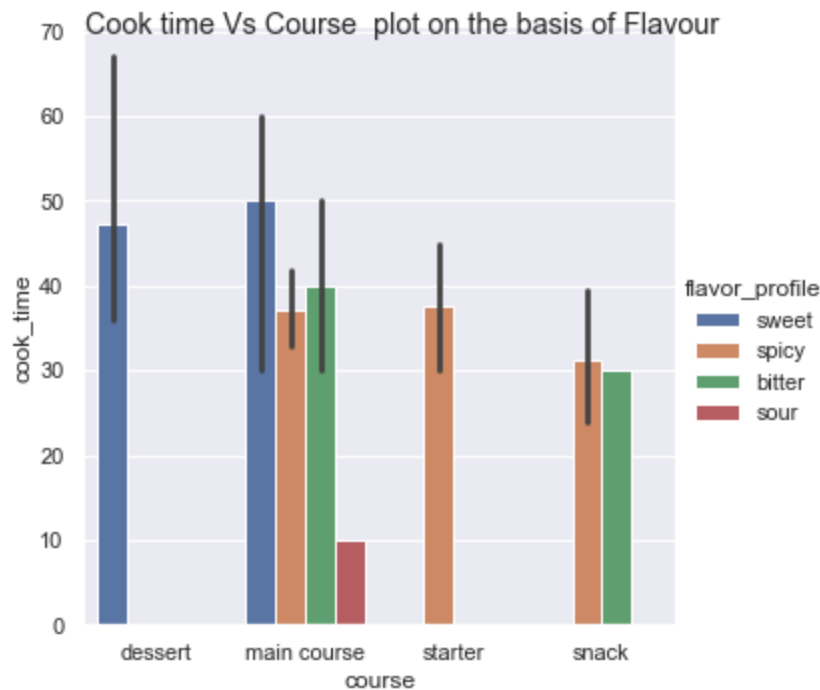
In [70]:

```
b=sns.catplot(data=df, kind="bar", x="course", y="cook_time", hue="flavor_profile")
b.fig.suptitle('Cook time Vs Course plot on the basis of Flavour')

d=sns.catplot(data=df, kind="bar", x="course", y="prep_time", hue="flavor_profile")
d.fig.suptitle('Cook time Vs diet plot on the basis of Flavour Profile')
```

Out[70]:

Text(0.5, 0.98, 'Cook time Vs diet plot on the basis of Flavour Profile')



Inference:

1) In main course:



The sweet flavoured dishes take longer to be prepared than to be cooked The bitter and the spicy dishes take longer to be cooked than to be prepared 2)In snacks:

The bitter snacks take longer to be cooked. The spicy snacks take longer to be prepared for.

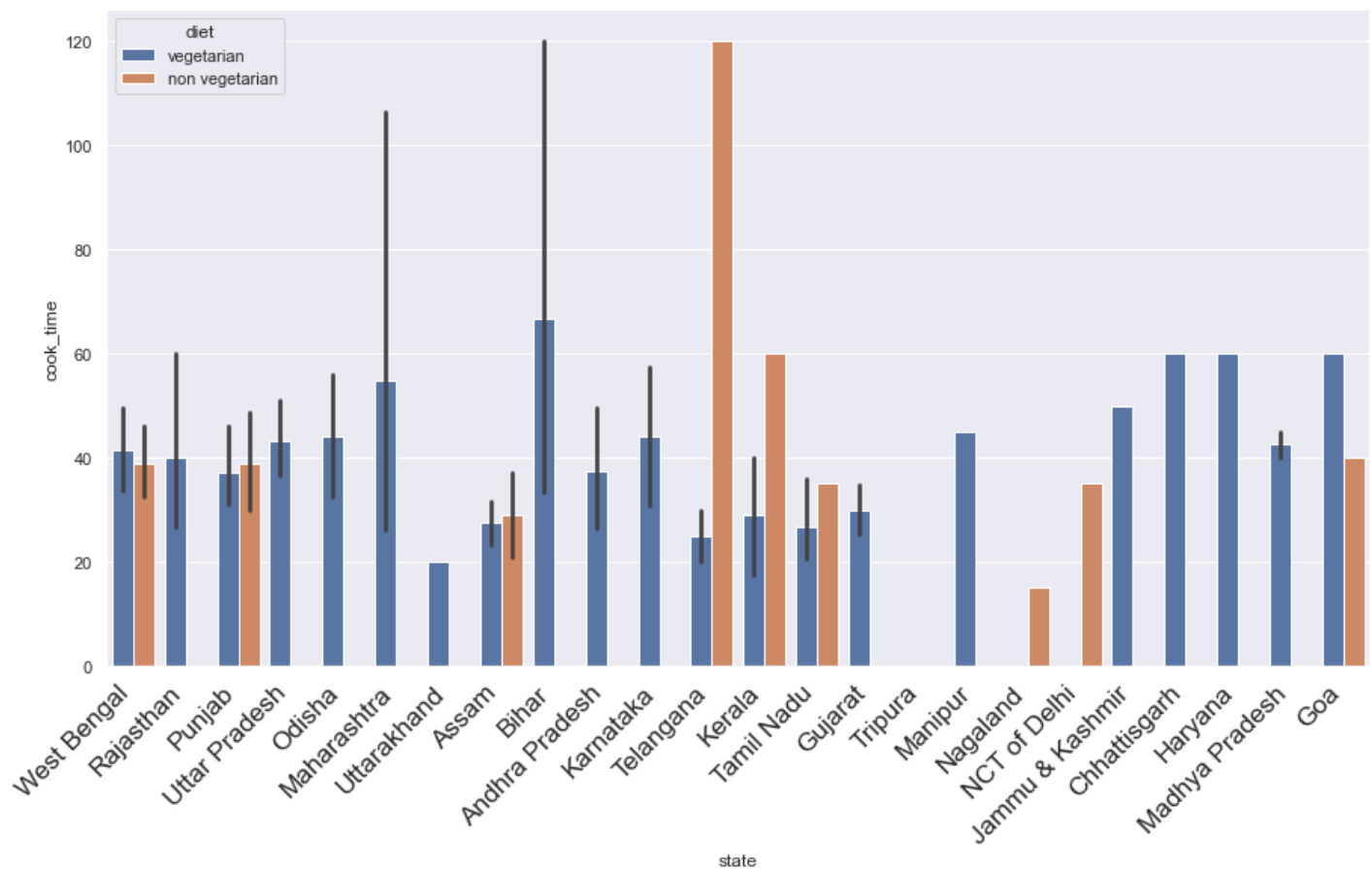
## 19. Compare cooking time and preparation time for dishes from certain state and of certain diet

In [71]:

```
#Statewise breakdown of diets
fig=plt.figure(figsize=(15,8))
sns.barplot(x='state',y='cook_time',hue='diet',data=df)
plt.xticks(
    rotation=45,
    horizontalalignment='right',
    fontweight='light',
    fontsize='x-large'
)
```

Out[71]:

```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
        17, 18, 19, 20, 21, 22, 23]),
 [Text(0, 0, 'West Bengal'),
  Text(1, 0, 'Rajasthan'),
  Text(2, 0, 'Punjab'),
  Text(3, 0, 'Uttar Pradesh'),
  Text(4, 0, 'Odisha'),
  Text(5, 0, 'Maharashtra'),
  Text(6, 0, 'Uttarakhand'),
  Text(7, 0, 'Assam'),
  Text(8, 0, 'Bihar'),
  Text(9, 0, 'Andhra Pradesh'),
  Text(10, 0, 'Karnataka'),
  Text(11, 0, 'Telangana'),
  Text(12, 0, 'Kerala'),
  Text(13, 0, 'Tamil Nadu'),
  Text(14, 0, 'Gujarat'),
  Text(15, 0, 'Tripura'),
  Text(16, 0, 'Manipur'),
  Text(17, 0, 'Nagaland'),
  Text(18, 0, 'NCT of Delhi'),
  Text(19, 0, 'Jammu & Kashmir'),
  Text(20, 0, 'Chhattisgarh'),
  Text(21, 0, 'Haryana'),
  Text(22, 0, 'Madhya Pradesh'),
  Text(23, 0, 'Goa')])
```

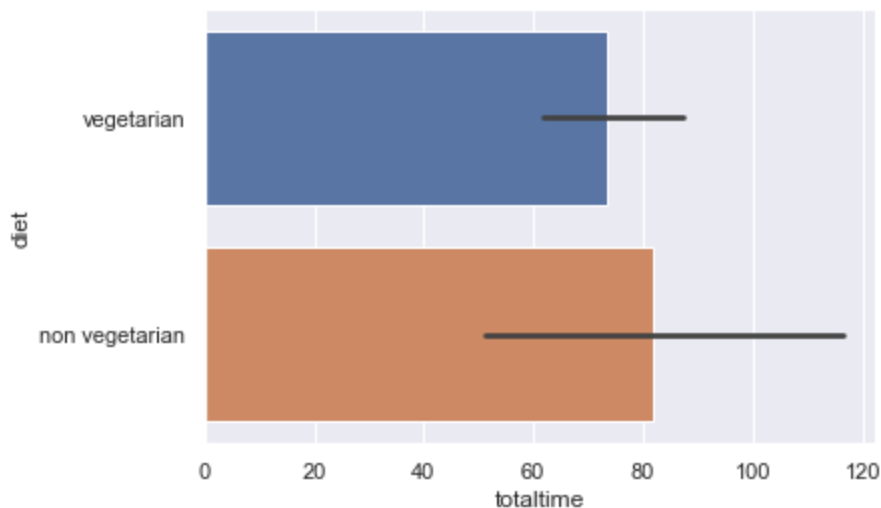


Inference: In rajasthan,uttar pradesh,odish,maharashtra,bihar,andhra pradesh,manipur,chandigarh,haryana,madhya pradesh mostly people in these states are vegetarian.

## 20. Display the total time taken to make every diet, to make every course, to make every flavor profile

```
In [72]: sns.barplot(x=df.totaltime,y=df.diet)
```

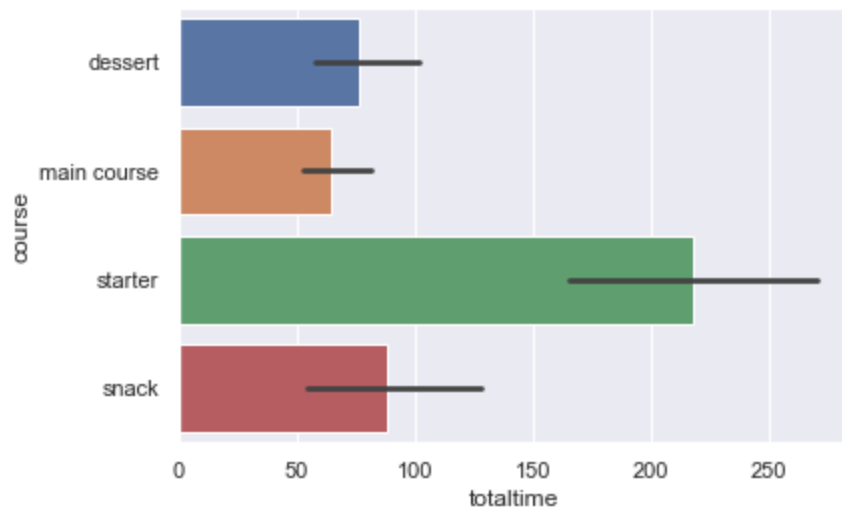
```
Out[72]: <AxesSubplot:xlabel='totaltime', ylabel='diet'>
```



Inference : We see that non vegetarian food takes longer to be made than vegetarian food.

```
In [73]: sns.barplot(x=df.totaltime,y=df.course)
```

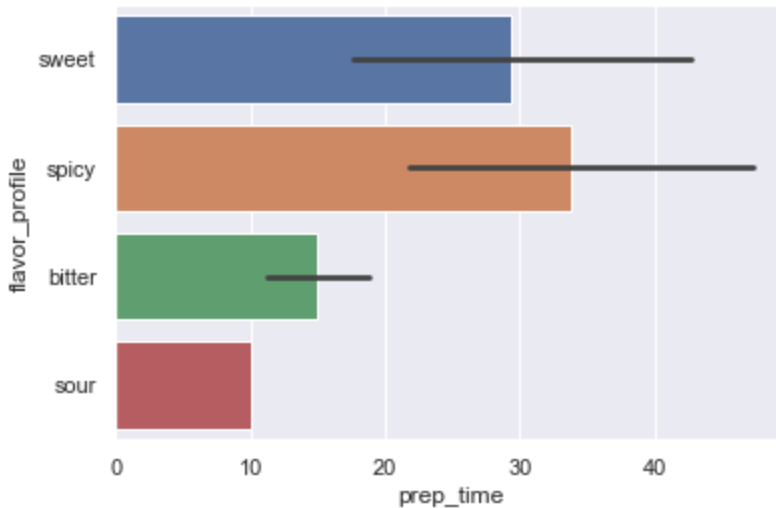
```
Out[73]: <AxesSubplot:xlabel='totaltime', ylabel='course'>
```



Inference : We see that the total time taken to make starters is longer than the time taken to make any other course.

In [74]: `sns.barplot(x=df.prep_time,y=df.flavor_profile)`

Out[74]: `<AxesSubplot:xlabel='prep_time', ylabel='flavor_profile'>`

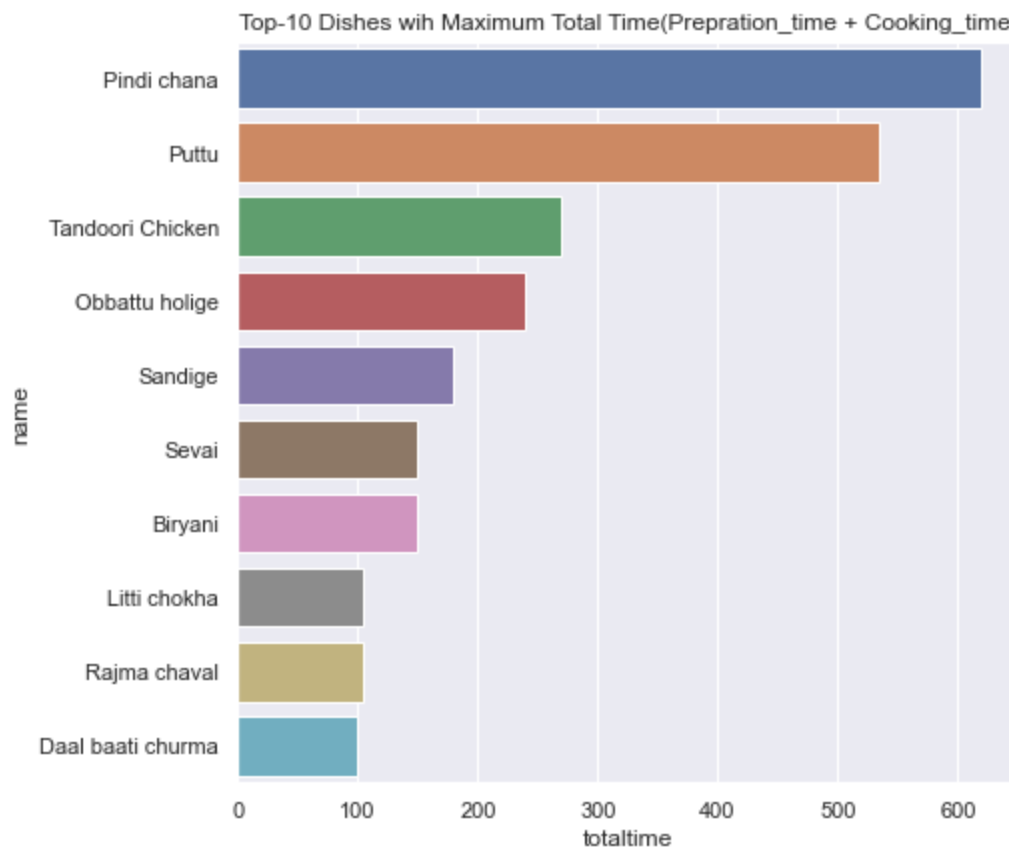


Inference: We see that it takes longer time to prepare sweet flavoured dishes than compared to other

## 21. Display the top 10 dishes in maincourse that take maximum totaltime.

In [75]: `down_main = df[df.course == "main course"].sort_values('totaltime', ascending = 0)[:10]  
plt.figure(figsize=(7,15))  
plt.subplot(2,1,1)  
sns.barplot(y = down_main.name, x = down_main.totaltime)  
plt.title("Top-10 Dishes wih Maximum Total Time(Prepration_time + Cooking_time)")`

Out[75]: `Text(0.5, 1.0, 'Top-10 Dishes wih Maximum Total Time(Prepration_time + Cooking_time)')`

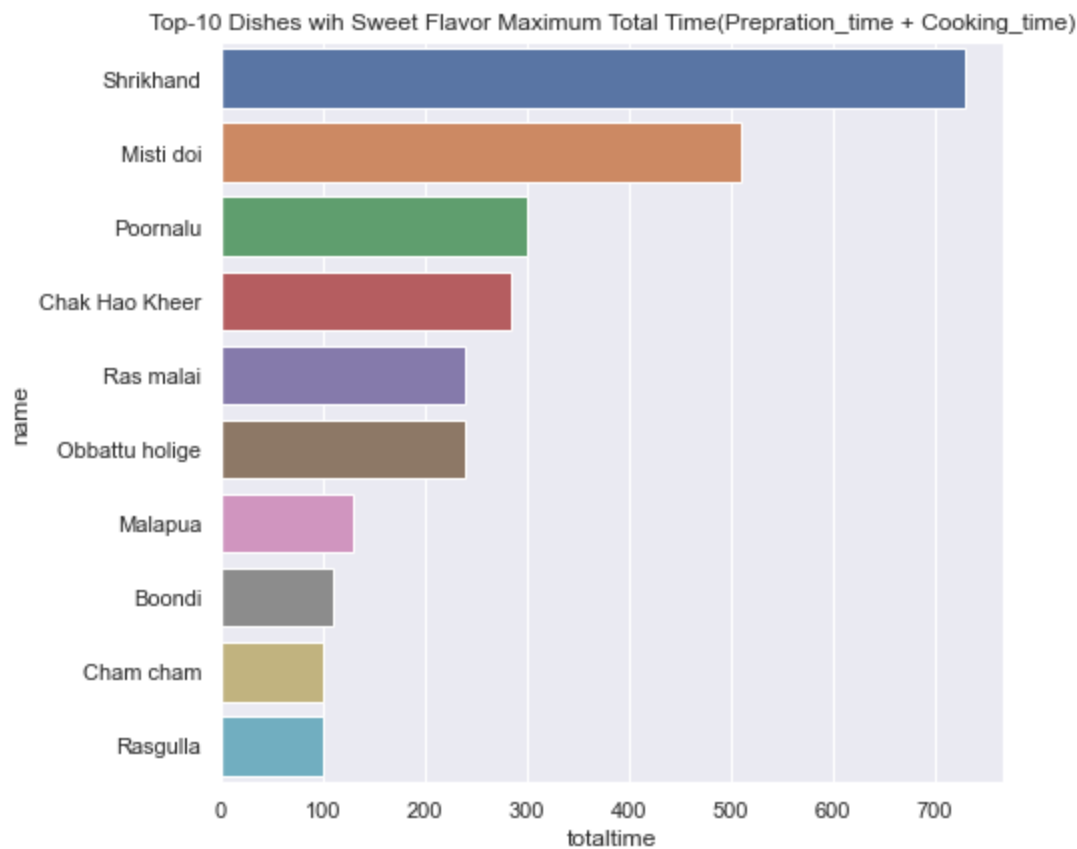


Inference : Pindi Channa is the maincourse dish that takes the longest to be made

## 22. Display the top 10 dishes in sweet flavor that take maximum totaltime

```
In [76]: down_main = df[df.flavor_profile == "sweet"].sort_values('totaltime', ascending = 0)[:10]
plt.figure(figsize=(7,15))
plt.subplot(2,1,1)
sns.barplot(y = down_main.name, x = down_main.totaltime)
plt.title("Top-10 Dishes wih Sweet Flavor Maximum Total Time(Prepration_time + Cooking_time)")

Out[76]: Text(0.5, 1.0, 'Top-10 Dishes wih Sweet Flavor Maximum Total Time(Prepration_time + Cooking_time)')
```

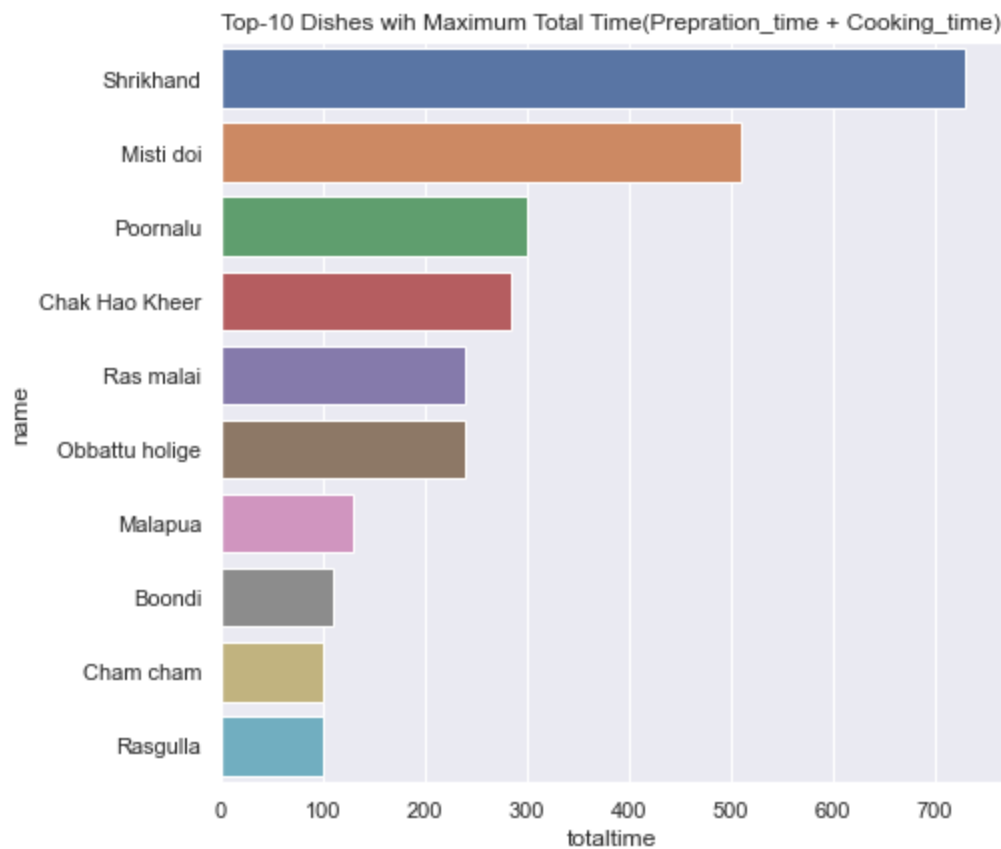


Inference: Shrikhand is the sweet flavoured dish that takes the longest to be made

## 23. Display the top 10 dishes with maximum total time in the entire dataset

```
In [77]: down_dishes = df.sort_values('totaltime', ascending = 0)[:10]
plt.figure(figsize=(7,15))
plt.subplot(2,1,1)
sns.barplot(y = down_main.name, x = down_main.totaltime)
plt.title("Top-10 Dishes with Maximum Total Time(Preparation_time + Cooking_time)")
```

```
Out[77]: Text(0.5, 1.0, 'Top-10 Dishes with Maximum Total Time(Preparation_time + Cooking_time)')
```



Inference: Shrikhand also takes the longest to be made from all the dishes in the data

## 24. Display the top 10 dishes with minimum total time in the entire dataset

In [78]: `df.nsmallest(10, 'totaltime')`

	name	ingredients	diet	prep_time	cook_time	flavor_profile	course	state	region	No_of_ing
11	Lassi	Yogurt, milk, nuts, sugar	vegetarian	5.0	5.0	sweet	dessert	Punjab	North	
111	Papad	Urad dal, sev, lemon juice, chopped tomatoes	vegetarian	5.0	5.0	spicy	snack	NaN	NaN	
147	Papadum	Lentils, black pepper, vegetable oil	vegetarian	5.0	5.0	spicy	snack	Kerala	South	
214	Khichu	Rice flour, sesame seeds, baking soda, peanut oil	vegetarian	5.0	10.0	spicy	snack	Gujarat	West	
109	Pani puri	Kala chana, mashed potato, boondi, sev, lemon	vegetarian	15.0	2.0	spicy	snack	NaN	NaN	

	name	ingredients	diet	prep_time	cook_time	flavor_profile	course	state	region	No_of_ingi
67	Galho	Rice, axone, salt, water, chillies, pork	non vegetarian	5.0	15.0	spicy	main course	Nagaland	North East	
78	Chapati	Whole wheat flour, olive oil, hot water, all p...	vegetarian	10.0	10.0	NaN	main course	Maharashtra	West	
169	Bajri no rotlo	Wheat flour, pearl millet flour, hot water	vegetarian	10.0	10.0	spicy	main course	Gujarat	West	
175	Chorafali	Urad dal, bengal gram flour, dried mango, baki...	vegetarian	5.0	15.0	spicy	snack	Gujarat	West	
190	Keri no ras	Mango, sugar	vegetarian	10.0	10.0	sour	main course	Gujarat	West	

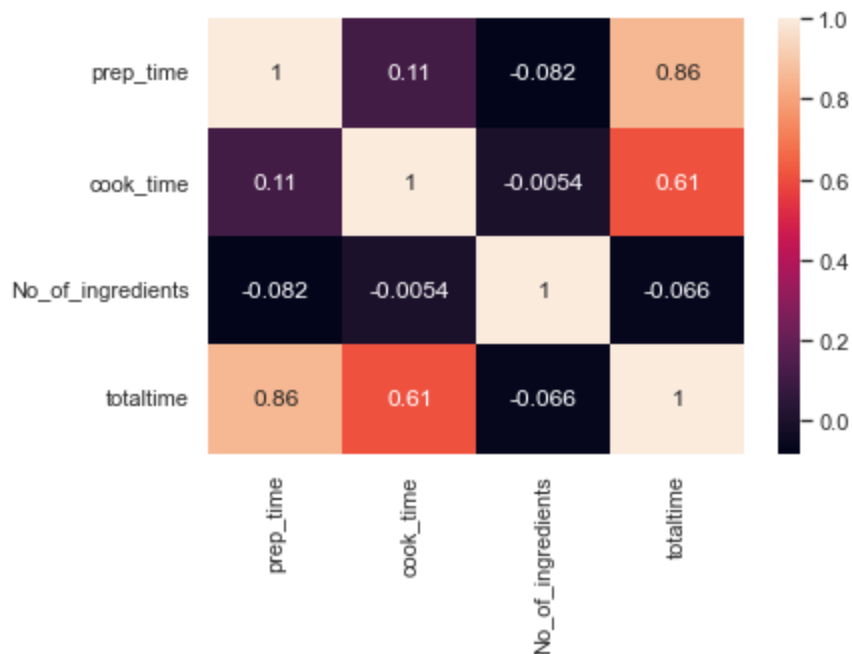
Inference: Lassi takes lowest time

## 25. Look for correlations between columns. Does it reveal anything? Can also use Heatmap to the show the relation between the numeric variables.

In [79]: `cor=df.corr()`

In [80]: `sns.heatmap(cor,annot=True)`

Out[80]: `<AxesSubplot:>`



## 26. What else can you understand from the data?

After having done the EDA we come to know the the distribution of various dishes through out the various states of the country and regions, as well as the total time taken to prepare it, the ingredients used in making them and their flavour profiles. Using this we can then build a model for an Indian food recommendation system at any restaurants or food delivery applications using the features total time, number of ingredients used, flavour profile and type of diet as primary determinants or filters.