Pick Up Time	Pick up location	...	Fare amount
1	Laguardia	...	$40
2	JFK	...	$30
...	...	...	...
...	...	...	...

Heap File

1	Laguardia	...	40	101	Times	...	10	1001	MET	...	15
2	JFK	...	30	102	ESB	...	31	1002	Park	...	20

Data Page 1 Data Page 2 ... Data Page N

Pick Up Time	Pick up location	...	Fare amount
1	Laguardia	...	$40
2	JFK	...	$30
102	ESB	...	$31
...	...	...	...

Heap File

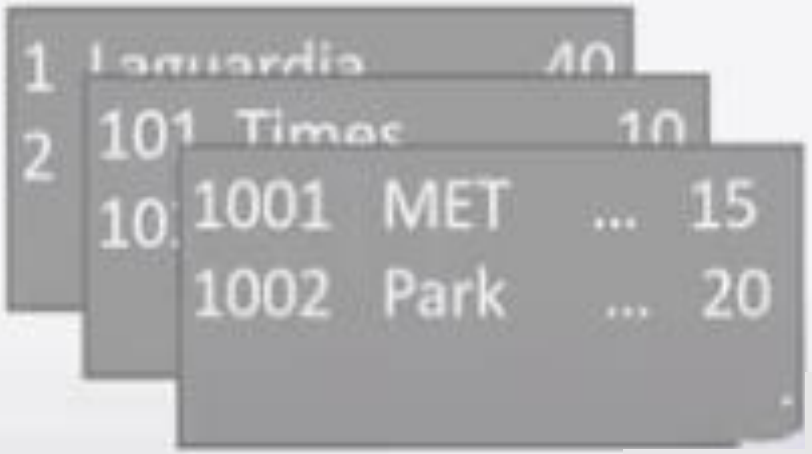
1	Laguardia	...	40	101	Times	...	10	...	1001	MET	...	15
2	JFK	...	30	102	ESB	...	31		1002	Park	...	20

Data Page 1 | Data Page 2 | Data Page N

Trip Time Stamp	Pick up location	....	Fare amount
1	Laguardia	...	$40
2	JFK	...	$30
3	<lat,long>	...	$...
...	...	...	...

Range Query

Return all the NYC taxi trips for which the pick up location is within the input range

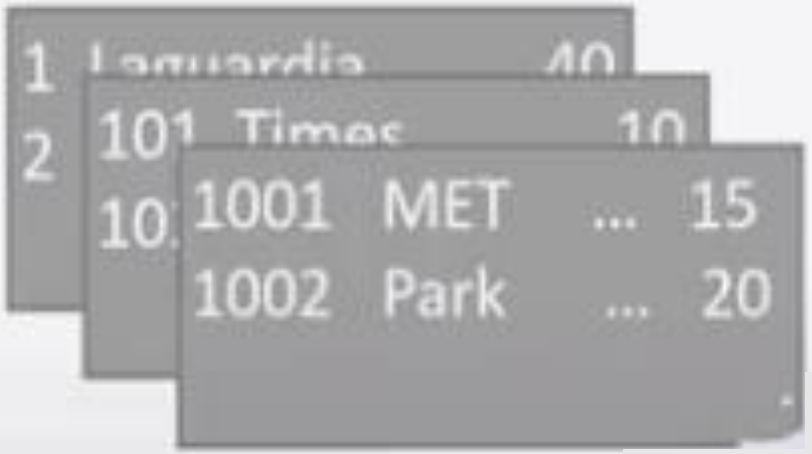


Data Pages

Trip Time Stamp	Pick up location	....	Fare amount
1	Laguardia	...	$40
2	JFK	...	$30
3	<lat,long>	...	$...
...	...	...	...

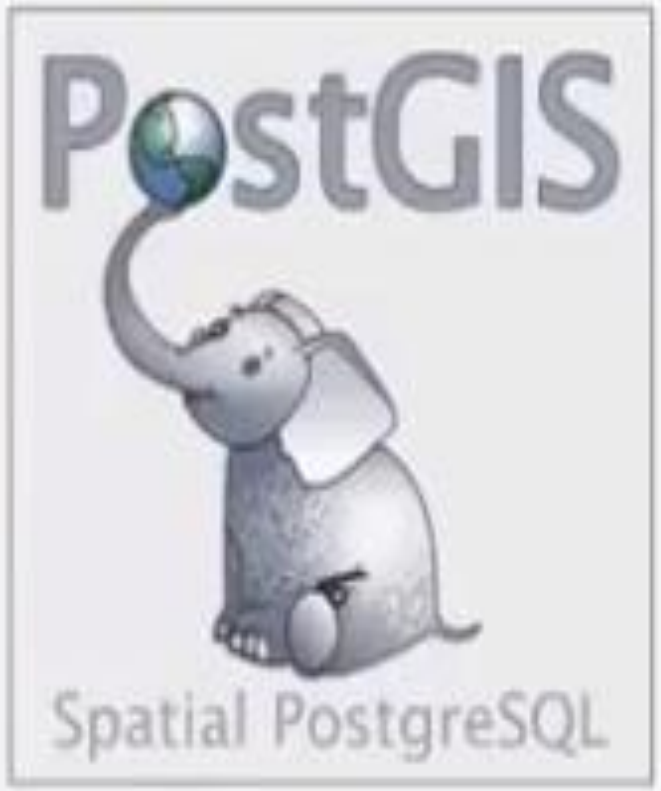
KNN Query

Return the **K** taxi trips for which the pick up location is the closest to Laguardia airport



Data Pages

Extend SQL to support Spatial Data



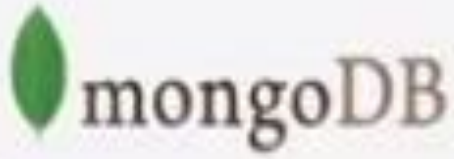

```
SELECT Restaurant.name

FROM city, restaurant

WHERE ST_Contains (city.geom, restaurant.geom)

AND city.name = 'Tempe'
```

Spatial data in NoSQL




```
var neighborhood = db.neighborhoods.findOne( {
geometry: { $geoIntersects: { $geometry: { type:
"Point", coordinates: [ -73.93414657,
40.82302903 ] } } } } )
```

```
db.restaurants.find( { location: { $geoWithin: {
$geometry: neighborhood.geometry } } } ).count()
```