Parking Lot

I own a multi-storey parking lot that can hold up to 'n' cars at any given point in time. Each slot is given an auto incrementing sequential number to uniquely identify itself. I want to create an automated ticketing system that allows my customers to use my parking lot without human intervention.

When a car enters my parking lot from any one of the multiple entry points, I want to have a ticket issued to the driver. The ticket issuing process includes us documenting the registration number (number plate) and the colour of the car and allocating an available parking slot to the car before actually handing over a ticket to the driver (we assume that our customers are nice enough to always park in the slots allocated to them). The customer should be allocated a parking slot which is nearest to the entry. At the exit the customer returns the ticket which then marks the slot they were using as being available.

Due to government regulation, the system should provide me with the ability to find out:

- Registration numbers of all cars of a particular colour.
- Slot number in which a car with a given registration number is parked.
- Slot numbers of all slots where a car of a particular colour is parked.

Parking lot landscape:

1. Multi storey parking.(Multiple lanes in single floor)
2. Vehicle categories(bike-4 per slot,car-2 slot,bus/truck-1 slot)
3. Entrypoint are selected slots present in 4 corner slots

Assumptions: there is enough space to move vehicles between slots.

Expectations :

- We are really interested in your low level design, object oriented skills so please keep that in mind.
- Please follow coding conventions and directory structure in whichever language you use.
- Input to your system can be done using a file, sysin or through test cases whichever way you want to choose.

Payment for parking based on time and category.(charges open to interpretation)

Sample Input / Output :

```
Input:
create_parking_lot 6 6

Output: Created a parking lot with 6x6 slots

Input: park KA-01-HH-1234  car White <entrypoint_id>

Output: Allocated slot
number: 1*1

Input: park KA-01-HH-9999  bike White
<entrypoint_id> 0

Output: Allocated slot
number: 1*2

Input: park KA-01-BB-0001 bike Black
<entrypoint_id> 0

Output: Allocated slot
number: 1*2

Input: park KA-01-HH-7777  car Red
<entrypoint_id> 0

Output: Allocated slot
number: 2*1

Input: park KA-01-HH-2701 car Blue
<entrypoint_id> 0

Output: Allocated slot
number: 2*1

Input: park KA-01-HH-3141 bus Black
<entrypoint_id> 4

Output: Allocated slot
number:  6*6


Input:
leave 1*2
```

Output: Slot number 1*2
is free with charges Rs
40 and time spent

Input:
Status

Output (Tab delimited output): No Registration
Slot No. Colour 1 KA-01-HH- 1234 car White 1*1
KA-01-HH- 9999 White 3 KA-01-BB- 0001 bike Black
1*2

KA-01-HH- 2701 Blue 6 KA-01-HH- 3141 bus Black
6*6

Other conditions to take care:

1.Parking lot is full

2. Slot query for invalid vehicle.

Note: Test cases mentioned above might not be valid according to the logic