

## PROJETO – SISTEMA DE LOCAÇÃO DE CARROS

### 1ª etapa (08/05 – 09/05 – 11/05) - Fazer menu e rotina para:

- Arquivo e Cadastro de carros contendo uma estrutura com:
  - nº de registro do carro (gerado automaticamente pelo sistema)
  - modelo
  - tipo: pequeno, médio ou grande
  - valor da diária
  - status irá conter a sigla de status, local de retirada e se for o caso: o registro do cliente, data de retirada, data e local de devolução
    - L – livre
    - A – alugado
    - R – reservado (data da retirada – 1 dia após a devolução do carro)
  - ❖ status deve ser **união (OR)**
  - ❖ definir o local de retirada de acordo com as lojas
  - ❖ cada carro poderá ter apenas 1 aluguel e 1 reserva
  
- Consulta de carros – 2 tipos de consulta:
  - **Total**
    - ❖ mostrar nº de registro, modelo, tipo e status (sigla, local de retirada e se for o caso: mostrar as tabelas)
  - **Parcial**
    - ❖ o usuário deverá entrar com a **data** da locação/reserva, o **tipo (P, M ou G)** e o **local** que deseja retirar
    - ❖ mostrar nº do registro, modelo e diária dos carros livres para a data, tipo e local escolhidos (se for o caso: mostrar a tabela de locação)

### Exemplo do Cadastro de Carros

Car	Modelo	Tipo	Diária	Status											
				Info_carro / Info_cliente						Info_carro / Info_cliente					
				Sigla	LRet	Cli	Ret	Dev	LDev	Sigla	LRet	Cli	Ret	Dev	LDev
1	Gol	P	100,00	L	SP	---	---	---	---	---	---	---	---	---	---
2	Golf	M	200,00	A	Sor	1	24/05	31/05	Sor	---	---	---	---	---	---
3	Uno	P	100,00	A	Sor	2	27/05	03/06	SP	R	SP	3	04/06	11/06	Sor
4	Corola	G	300,00	L	SP	---	---	---	---	---	---	---	---	---	---

```
typedef struct carro{
    int          reg_car;           // gerado automaticamente
    char         modelo[20];
    char         tipo;             // [P]equeno, [M]édio ou [G]rande
    float        diaria;
    union estado status;
} carro;
```

```
union      estado{
    struct info_carro  car;
    struct info_cliente dados[2];
};
```

```
struct      info_carro{
    char      sigla;           // L – Livre
    char      local_ret[20];   //local da retirada
};
```

```
struct      info_cliente{
    char      sigla;           // A – Alugado ou R – Reservado
    char      local_ret[20];   //local da retirada
    int       reg_cli;         // nº cliente – buscar no arq. cliente
    int       dia_ret;         // dia – máx=31
    int       mes_ret;         // mes: fev=28 dias
    int       dia_dev;         // mes: abril=jun=set=nov=30 dias
    int       mes_dev;         // os demais = 31 dias
    char      local_dev[20];   //local da devolução
};
```