

3) Crie em SQL e execute as seguintes operações:

a) Criação de todas as tabelas.

```
CREATE TABLE `alugar` (  
  `cpf_cliente` varchar(11) NOT NULL,  
  `isbn_livro` varchar(20) NOT NULL,  
  `dtHora_aluga` date NOT NULL,  
  `dtDevolucao_aluga` date NOT NULL,  
  PRIMARY KEY (`cpf_cliente`, `isbn_livro`, `dtHora_aluga`),  
  KEY `fk_isbn_livro_aluga_idx` (`isbn_livro`),  
  CONSTRAINT `fk_cpf_cliente_aluga` FOREIGN KEY (`cpf_cliente`) REFERENCES `clientes`  
  (`cpf_cliente`),  
  CONSTRAINT `fk_isbn_livro_aluga` FOREIGN KEY (`isbn_livro`) REFERENCES `livros`  
  (`cod_livro`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
CREATE TABLE `clientes` (  
  `cpf_cliente` varchar(11) NOT NULL,  
  `dtNascimento_cliente` date DEFAULT NULL,  
  `nome_cliente` varchar(45) NOT NULL,  
  `endereco_cliente` varchar(45) DEFAULT NULL,  
  `telefone_cliente` varchar(45) NOT NULL,  
  PRIMARY KEY (`cpf_cliente`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
CREATE TABLE `comprar` (  
  `cpf_cliente` varchar(11) DEFAULT NULL,  
  `dtCompra_compra` date DEFAULT NULL,  
  `notaFiscal_compra` varchar(20) NOT NULL,  
  PRIMARY KEY (`notaFiscal_compra`),  
  KEY `fk_cpf_cliente_compra` (`cpf_cliente`),  
  CONSTRAINT `fk_cpf_cliente_compra` FOREIGN KEY (`cpf_cliente`) REFERENCES  
  `clientes` (`cpf_cliente`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
CREATE TABLE `itensCompra` (  
  `notafiscal_compra` varchar(20) NOT NULL,  
  `codLivro_livro` varchar(20) NOT NULL,  
  `quantidade_itensCompra` int(11) NOT NULL,  
  PRIMARY KEY (`notafiscal_compra`, `codLivro_livro`),  
  KEY `fk_cod_livro_itensCompra_idx` (`codLivro_livro`),  
  CONSTRAINT `fk_cod_livro_itensCompra` FOREIGN KEY (`codLivro_livro`) REFERENCES  
  `livros` (`cod_livro`) ON DELETE NO ACTION ON UPDATE NO ACTION,  
  CONSTRAINT `fk_notafiscal_compra_itensCompra` FOREIGN KEY (`notafiscal_compra`)  
  REFERENCES `comprar` (`notaFiscal_compra`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
CREATE TABLE `livros` (
  `cod_livro` varchar(20) NOT NULL,
  `nome_livro` varchar(45) NOT NULL,
  `descricao_livro` varchar(45) DEFAULT NULL,
  `qtdEstoque_livro` int(11) DEFAULT NULL,
  `modalidade_livro` int(11) NOT NULL,
  `editora_livro_livro` varchar(45) NOT NULL,
  `anoPublicacao_livro` int(11) NOT NULL,
  `tipo_livro` int(11) NOT NULL,
  `preco_livro` float NOT NULL,
  PRIMARY KEY (`cod_livro`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

b) Emissão de relatório contendo os livros mais vendidos em um determinado mês;

```
select nome_livro as 'Nome do Livro', quantidade_itensCompra as 'Quantidade'
      from itensCompra,comprar,livros
      where itensCompra.notafiscal_compra = comprar.notaFiscal_compra and
            cod_livro = codLivre_livro and
            month(dtCompra_compra) = '04'
      order by quantidade_itensCompra DESC;
```

c) Emissão de relatório de livros menos vendidos em um determinado mês;

```
select nome_livro as 'Nome do Livro', quantidade_itensCompra as 'Quantidade'
      from itensCompra,comprar,livros
      where itensCompra.notafiscal_compra = comprar.notaFiscal_compra and
            cod_livro = codLivre_livro and
            month(dtCompra_compra) = '04'
      order by quantidade_itensCompra;
```

d) Emissão de relatório de livros alugados em um determinado mês;

```
select nome_livro as 'Nome do Livro'
      from alugar,clientes,livros
      where clientes.cpf_cliente = alugar.cpf_cliente and
            alugar.isbn_livro = livros.cod_livro and
            month(alugar.dtHora_aluga) = '01';
```

e) Quanto foi faturado mensalmente com o aluguel de livros;

```
select sum((((datediff(dtDevolucao_aluga, alugar.dtHora_aluga))-7)*2) as multa,
      month(dtHora_aluga) as mes from alugar
      where datediff(dtDevolucao_aluga,
alugar.dtHora_aluga) > 7 group by mes ;
```

f) Quanto foi faturado mensalmente com a venda de livros;

```
select MONTH(comprar.dtCompra_compra) as mes , sum(livros.preco_livro) from
comprar, itensCompra, livros where itensCompra.codLivro_livro = livros.cod_livro
and itensCompra.notafiscal_compra = comprar.notaFiscal_compra group by mes ;
```

g) Qual o mês de menor faturamento.

```
select mes, (sum(faturamento)) as mensal from (
    select MONTH(comprar.dtCompra_compra) as mes , sum(livros.preco_livro) as
faturamento from
    itensCompra, livros, alugar right JOIN comprar
    ON alugar.dtDevolucao_aluga = comprar.dtCompra_compra
    where itensCompra.codLivro_livro = livros.cod_livro and
    itensCompra.notafiscal_compra = comprar.notaFiscal_compra
    group by mes
    UNION ALL
    select month(dtHora_aluga) as mes , sum((((datediff(dtDevolucao_aluga,
alugar.dtHora_aluga))-7)*2) as faturamento from
    alugar left JOIN comprar
    ON alugar.dtDevolucao_aluga = comprar.dtCompra_compra
    where datediff(dtDevolucao_aluga, alugar.dtHora_aluga) > 7
    group by mes
) derivedTable
    group by mes
ORDER BY MENSAL
LIMIT 1;
```