

# **SISTEMA DE REGISTRO DE HÓSPedes E RESERVAS E CONHECIMENTOS GERADOS A PARTIR DE SEUS DADOS**

## **1 - FINALIDADE DO PROJETO**

Este projeto foi desenvolvido como objeto de enriquecimento de entrevista e prova de experiência para a vaga de Analista de Suporte na empresa Prion Software.

## **2 – HABILIDADES NECESSÁRIAS PARA O DESENVOLVIMENTO DO PROJETO**

- I – Banco de Dados
- II – SQL
- III – GIT
- IV – HTML
- V – CSS
- VI – PHP

Propus-me a desenvolver um projeto que tivesse como requisitos todas as habilidades propostas pela vaga de Analista de Suporte na Prion.

Como é de notório saber, não foi utilizado a linguagem C# no projeto (mesmo esta sendo um requisito da vaga), porém este conhecimento está sendo desenvolvido por mim a parte.

## **3 – IDEALIZAÇÃO DO PROJETO**

A idealização deste projeto consiste na criação e gerenciamento de dois formulários de registro de dados de um empreendimento Hoteleiro, "Pousada X".

O primeiro formulário, Cadastro de Hóspede, visa registrar os dados do cliente que efetuou a reserva.

Já o segundo formulário, Informação da Reserva, tem função de, ao fim da reserva, registrar dados relevantes a respeito do cliente: número de diárias e qualificação de como o hóspede se portou durante sua estadia.

Desse jeito, é possível auferir informações de negócio, possibilitando o conhecimento e, logo, a tomada de decisão.

## **4 - O PROJETO**

### **4.1 - BANCO DE DADOS E TABELAS**

O banco de dados e suas respectivas tabelas foram criadas e gerenciadas através da linguagem SQL no MySQL.

Abaixo serão registrados o passo-a-passo dos comandos usados para o desenvolvimento do banco de dados.

## Criação do Banco de Dados

Para início do projeto, foi criado um banco de dados chamado “pousada”.

The screenshot shows the MySQL Workbench interface. The top menu bar has tabs for 'Base de Dados', 'SQL', 'Estado', and 'Cont'. The 'Base de Dados' tab is selected. Below the tabs, there is a message box that says 'Executar consulta(s) SQL no servidor "127.0.0.1":'. In the main query editor area, there are two numbered lines of SQL code:

```
1 CREATE DATABASE pousada
2
```

## Criação das Tabelas

Logo, duas tabelas foram criadas para substanciar o banco de dados. São elas:

*Tabela Hóspede*

The screenshot shows the MySQL Workbench interface for the 'hospede' table. The top menu bar has tabs for 'Procurar', 'Estrutura', 'SQL', and 'Pesquisar'. The 'SQL' tab is selected. Below the tabs, there is a message box that says 'Run SQL query/queries on table pousada.hospede:'. In the main query editor area, there are eight numbered lines of SQL code:

```
1 CREATE TABLE hospede
2 (`CPF` CHAR(11) NOT NULL PRIMARY KEY,
3 `Nome Completo` VARCHAR(50) NOT NULL,
4 `Endereço` VARCHAR(50),
5 `Cidade` VARCHAR(50),
6 `Estado` CHAR(2) NOT NULL,
7 `Sexo` CHAR(1),
8 `Data de Nascimento` DATE NOT NULL)
```

*Tabela Reserva*

The screenshot shows the MySQL Workbench interface for the 'reserva' table. The top menu bar has tabs for 'Procurar', 'Estrutura', 'SQL', and 'Pesquisar'. The 'SQL' tab is selected. Below the tabs, there is a message box that says 'Run SQL query/queries on table pousada.reserva:'. In the main query editor area, there are seven numbered lines of SQL code:

```
1 CREATE TABLE reserva
2 (`ID da reserva` SMALLINT AUTO_INCREMENT PRIMARY KEY,
3 `CPF Hóspede` CHAR(11),
4 `Check-in` DATE,
5 `Check-out` DATE,
6 `Número de diárias` SMALLINT,
7 `Qualificação do Hóspede` VARCHAR(7))
```

## **4.2 – ARQUIVOS DE DESENVOLVIMENTO**

O software “XAMPP Control Panel”, através do MySQL, foi usado como responsável pela criação de um banco de dados interno a fim de armazenar os dados fornecidos pelos formulários desenvolvidos. E, através do “Apache”, foi concedido suporte à linguagem PHP, usada na comunicação com o banco de dados.

A fim de elaborar os sistemas de registros de dados, foram desenvolvidos arquivos que dessem suporte às funcionalidades exigidas. São eles:

### ***I – index.html e estilo.css (Página Inicial)***

Estes foram os responsáveis pela página inicial do projeto, contendo duas opções de navegação:

- a) Cadastro do Hóspede
- b) Informação da Reserva



### ***II – cad-hospede.php e cad-hospede.css (Cadastro de Hóspede)***

Estes foram os responsáveis por desenvolver o formulário do “Cadastro de Hóspede”, que tem como função capturar os dados, como: CPF, Nome, Endereço, Cidade, Estado, Sexo e Data de Nascimento, e salvá-los no banco de dados “pousada”, na tabela “hospede”.

Cadastro de Hóspede - Projeto

localhost/html-css/cad-hospede.php

## CADASTRO DE HÓSPEDE

CPF  
Digite o CPF

Nome  
Digite o Nome Completo

Endereço  
Digite o Endereço

Cidade  
Digite a Cidade

Estado  
Digite a sigla do Estado

Sexo  
Masculino

Data de Nascimento  
dd/mm/aaaa

ENVIAR

### III – cad-reserva.php e cad-reserva.css (Informação da Reserva)

Estes também, como os arquivos anteriores, têm a função de capturar os dados e destiná-los ao banco de dados “pousada”, agora, na tabela “reserva”. São dados a serem inseridos no formulário “Informação da Reserva”: CPF do cliente, data do Check-in, Check-out e Qualificação do Hóspede, que pode ser definida como “bom”, “regular” ou “ruim”.

Informação da Reserva - Projeto

localhost/html-css/cad-reserva.php

## INFORMAÇÃO DA RESERVA

CPF do Reservante  
Digite o CPF do cliente

Data da Reserva

Check-in:  
dd/mm/aaaa

Check-out:  
dd/mm/aaaa

Qualificação do Hóspede  
Bom

ENVIAR

#### **IV – processa-hospede.php e processa-reserva.php**

Estes arquivos têm a função de fazer a captura dos dados imputados pelo usuário, tanto da página “Cadastro do Hóspede” quanto da página “Informação da Reserva”, destinando-os, através da linguagem PHP, às tabelas “hospede” e “reserva”, respectivamente.

#### **V – conexao-hospede e conexao-reserva**

Ambos arquivos têm função de fazer o link entre o banco de dados MySQL com os arquivos de captura, citados anteriormente.

#### **4.3 - INSERÇÃO DOS DADOS**

Após feita a criação do banco de dados e suas respectivas tabelas, foi possível conectá-los com o sistema de input PHP. Dessa forma, a inserção dos dados foi feita de forma sistêmica, procurando produzir um cenário mais real possível.

No registro “Cadastro de Hóspede”, por exemplo, foram inseridos 73 hóspedes com CPF único, nomes próprios de todas as letras do alfabeto, de regiões, sexo e data de nascimento diferentes.

	CPF	Nome Completo	Endereço	Cidade	Estado	Sexo	Data de Nascimento
<input type="checkbox"/>	Edita Copiar Apagar	00022266698 Thamires Tomas	Rua Guarani	Piracicaba	SP	F	1992-05-01
<input type="checkbox"/>	Edita Copiar Apagar	00159753864 Marcos Martins	Rua Pereira Nunces	Rio de Janeiro	RJ	M	1997-05-29
<input type="checkbox"/>	Edita Copiar Apagar	00236985214 Dorgival Dantas	Rua da Liguica	Armacao dos Buzios	RJ	M	1994-07-28
<input type="checkbox"/>	Edita Copiar Apagar	00256497853 Lucas Lottus	Rua Goiabeira	Feira de Santana	BA	M	1989-01-31
<input type="checkbox"/>	Edita Copiar Apagar	00302558945 Josias Jesus	Av. Presidente Vargas	Rio de Janeiro	RJ	M	1985-05-05
<input type="checkbox"/>	Edita Copiar Apagar	00325896965 Caio Conde	Av. Paraisopolis	Sao Jose dos Campos	SP	M	1992-05-25
<input type="checkbox"/>	Edita Copiar Apagar	00345122981 Renato Romero	Rua Reis Magos	Montes Brancos	BA	M	1975-08-09
<input type="checkbox"/>	Edita Copiar Apagar	00564278944 Fernanda Fernandes	Rua Sebastiao Calegari	Sao Lucas dos Pinhais	BA	F	1989-01-21
<input type="checkbox"/>	Edita Copiar Apagar	00614689566 Gediel Guimaraes	Av. Ribeiro Dantas	Contagem	MG	M	2010-09-19
<input type="checkbox"/>	Edita Copiar Apagar	01556895522 Alvaro Amancio	Rua Rosa dos Ventos, 5 Tatui		GO	M	1978-06-23

Para a tabela “Informação da Reserva” procurou-se seguir o mesmo critério, porém neste, o número de registros foi de 77.

	ID da Reserva	CPF Hospede	Check-in	Check-out	Numero de diárias	Qualificacao do Hospelde
<input type="checkbox"/>	Edita Copiar Apagar	14 00022266698	2022-01-01	2022-01-14	13	Bom
<input type="checkbox"/>	Edita Copiar Apagar	15 00159753864	2022-01-01	2022-01-06	5	Bom
<input type="checkbox"/>	Edita Copiar Apagar	16 00236985214	2022-01-02	2022-01-06	4	Regular
<input type="checkbox"/>	Edita Copiar Apagar	17 00256497853	2022-01-04	2022-01-20	16	Bom
<input type="checkbox"/>	Edita Copiar Apagar	18 00345122981	2022-01-08	2022-01-12	4	Ruim
<input type="checkbox"/>	Edita Copiar Apagar	19 00564278944	2022-01-09	2022-01-13	4	Regular
<input type="checkbox"/>	Edita Copiar Apagar	20 00614689566	2022-01-16	2022-01-22	6	Bom
<input type="checkbox"/>	Edita Copiar Apagar	21 01556895522	2022-01-16	2022-01-25	9	Ruim
<input type="checkbox"/>	Edita Copiar Apagar	22 02587419639	2022-01-20	2022-01-24	4	Bom
<input type="checkbox"/>	Edita Copiar Apagar	23 02589967415	2022-01-21	2022-01-29	8	Regular

#### **4.4 - SIMULAÇÃO DE CORREÇÕES EM DADOS DO BANCO DE DADOS**

A fim de simular uma correção nos dados inseridos erroneamente, foram propostas duas situações:

- I - O cliente "Otavio Onix" relatou que seu endereço estava errado. E que, na verdade, morava na "Rua Bueno Brandao", diferente do que foi inserido no seu registro, "Rua Monte Arara".

Primeiro registro do cliente Otavio Onix que, erroneamente, tinha como endereço a "Rua Monte Arara".

02587419639	Otavio Onix	Rua Monte Arara	Barbacena	MG	M	2001-01-12
-------------	-------------	-----------------	-----------	----	---	------------

Comandos para mudança de endereço.

The screenshot shows the MySQL Workbench interface with the SQL tab selected. The query window contains the following SQL code:

```
1 UPDATE hospede
2 SET Endereco = 'Bueno Brandao'
3 WHERE CPF = 02587419639
```

Resultado correto, atualizando o endereço "Rua Bueno Brandao".

02587419639	Otavio Onix	Bueno Brandao	Barbacena	MG	M	2001-01-12
-------------	-------------	---------------	-----------	----	---	------------

- II - A cliente "Samira Souza" relatou que sua data de nascimento estava errada. E que, na verdade, nascera em 03-10-1998.

Primeiro registro da cliente Samira Souza mostra sua data de nascimento como "03/10/1996".

09638852487	Samira Souza	Rua Gomes Leite	Curitiba	PR	F	1996-10-03
-------------	--------------	-----------------	----------	----	---	------------

Usou-se os seguintes comandos para realizar a mudança de data de nascimento:

The screenshot shows the MySQL Workbench interface with the SQL tab selected. The query window contains the following SQL code:

```
1 UPDATE hospede
2 SET `Data de Nascimento` = '1998-10-03'
3 WHERE CPF = 09638852487
```

Logo, tem-se o resultado correto:

09638852487 Samira Souza Rua Gomes Leite Curitiba PR F 1998-10-03

#### **4.5 - TOMADA DE CONHECIMENTO**

A partir de todo acúmulo de dados obtidos, é possível tirar algumas conclusões que podem servir como norte para o gestor da “Pousada X”, que poderá usar tais informações a favor de seu negócio.

Portanto, listei seis conhecimentos de negócio que podem ajudar na progressão da “Pousada X”. Para isso, mostrarei os comandos usados e os resultados gerados.

São eles:

##### **I - Qual sexo se hospeda mais na “Pousada X”?**

A partir do comando abaixo, obteve-se o resultado:

```
1 SELECT `Sexo`, COUNT(`Sexo`) AS `Quantidade`
2 FROM hospede GROUP BY Sexo
```

Sexo	Quantidade
F	28
M	45

**Conclusão:** É possível inferir que pessoas do sexo masculino são a maioria dos clientes, compondo cerca de 61% dos hóspedes do ano de 2022.

##### **II - Quantos são os hóspedes entre 10-20 anos, +20-30 anos, +30-40 anos, +40-50 anos, +50-60 anos e +60 anos?**

###### **Hóspedes entre 10 e 20 anos.**

A partir do comando abaixo, foi possível listar as pessoas entre 10 e 20 anos.

```
SELECT `Nome Completo`, `Data de Nascimento`
FROM hospede
WHERE `Data de Nascimento` > '2012-01-01' OR `Data de Nascimento` > '2002-01-01'
```

Tendo o resultado:

Nome Completo	Data de Nascimento
Gediel Guimaraes	2010-09-19
Fabiola Freitas	2006-01-23
Sabrina Santos	2009-12-02
Rafael Ragazzo	2008-09-02
Larissa Lorenzo	2012-01-05
Henrique Homero	2003-03-03
Deize Damiao	2003-03-03
Ubiratan Urca	2008-08-08
Davi Damasio	2006-02-25
Natalia Nobre	2009-08-15

```
SELECT COUNT(`Data de Nascimento`) AS `Pessoas entre 10 e 20 anos`
FROM hospede
WHERE `Data de Nascimento` > '2012-01-01' OR `Data de Nascimento` > '2002-01-01'
```

### Pessoas entre 10 e 20 anos

10

### Hóspedes entre 20 e 30 anos.

Utilizando, agora, o comando 'BETWEEN' para verificar as idades entre 20 e 30 anos.

```
1 SELECT `Nome Completo`, `Data de Nascimento`
2 FROM hospede
3 WHERE `Data de Nascimento` BETWEEN '1992-12-31' AND '2002-12-31'
```

Temos o resultado:

Nome Completo	Data de Nascimento
Marcos Martins	1997-05-29
Dorgival Dantas	1994-07-28
Otavio Onix	2001-01-12
Samira Souza	1998-10-03
Guilhermino Graca	1994-10-01
Kevin Korth	1999-12-06
Lucas Leite	1996-11-21
Denis Dante	1999-05-26
Raimundo Rodrigues	2001-05-05
Vanessa Vidal	1994-08-07
Juan Jamil	1995-12-20
Julio Jarbas	2000-11-30
Jessica Jurema	2000-10-01

Blenda Borel	2001-11-06
Matheus Mureb	1993-10-01
Vagner Voltz	1999-02-02
Zemira Zilda	1995-03-02
Rodrigo Ramires	1993-11-02
Natalia Normam	2001-05-05
Henry Honda	1994-06-18
Beatriz Bueno	2000-09-08
Henry Honario	2000-05-26
Lais Lamuriel	1998-02-20
Adelino Amadeu	1998-12-25
Raphael Reis	1999-12-06

Fazendo o somatório, temos:

```
1 SELECT COUNT(`Data de Nascimento`) AS `Hóspede entre 20 e 30 anos`  
2 FROM hospede  
3 WHERE `Data de Nascimento` BETWEEN '1992-12-31' AND '2002-12-31'  
4
```

Hóspedes entre 20 e 30 anos  
29

### **Hóspedes entre 30 e 40 anos.**

Segue-se a mesma lógica:

```
1 SELECT `Nome Completo`, `Data de Nascimento`  
2 FROM hospede  
3 WHERE `Data de Nascimento` BETWEEN '1982-12-31' AND '1992-12-31'
```

Logo, tem-se o resultado:

Nome Completo	Data de Nascimento
Thamyres Tomas	1992-05-01
Lucas Lottus	1989-01-31
Josias Jesus	1985-05-05
Caio Conde	1992-05-25
Fernanda Fernandes	1989-01-21
Caio Couto	1991-12-05
Almir Amaral	1990-07-28
Carlos Comenico	1988-01-30
Joao Junior	1992-06-03
Felipe Foham	1988-01-06
Vanessa Venancio	1991-02-05
Juliana Juma	1987-07-07
Evanildo Espertiz	1991-07-25
Savio Santos	1987-06-09
Iago Iglesias	1985-01-02
Jamile Jordana	1989-05-02
Elias Elpideo	1985-02-02
Fernanda Fontes	1990-06-06
Jonatas Junior	1987-06-29
Luiza Letrim	1990-02-02

Fazendo o somatório:

```
1 SELECT COUNT(`Data de Nascimento`) AS `Hóspede entre 30 e 40 anos`  
2 FROM hospede  
3 WHERE `Data de Nascimento` BETWEEN '1982-12-31' AND '1992-12-31'  
4
```

Hóspede entre 30 e 40 anos  
20

## Hóspedes entre 40 e 50 anos

```
1 SELECT `Nome Completo`, `Data de Nascimento`
2 FROM hospede
3 WHERE `Data de Nascimento` BETWEEN '1972-12-31' AND '1982-12-31'
```

Nome Completo	Data de Nascimento
Renato Romero	1975-08-09
Alvaro Amancio	1978-06-23
Paula Pontes	1980-05-06
Vanessa Voltz	1980-01-18
Matheus Moratti	1980-05-10
Jose Jamil	1980-09-05
Daiane Damares	1982-03-06
Gilmar Guimaraes	1979-09-17
Genilda Gentil	1980-04-25

Logo, faz-se o somatório.

```
1 SELECT COUNT(`Data de Nascimento`) AS `Hóspedes entre 40 e 50 anos`
2 FROM hospede
3 WHERE `Data de Nascimento` BETWEEN '1972-12-31' AND '1982-12-31'
```

Hóspedes entre 40 e 50 anos

9

## Hóspedes entre 50 e 60 anos

Usando os comandos abaixo:

```
1 SELECT `Nome Completo`, `Data de Nascimento`
2 FROM hospede
3 WHERE `Data de Nascimento` BETWEEN '1962-12-31' AND '1972-12-31'
```

Tem-se como resultado:

Nome Completo	Data de Nascimento
Giulia Guimaraes	1963-08-28
Luis Lorival	1963-09-19
Queiroz Queiroga	1969-06-08
Carlos Couto	1969-06-25
Juarino Junior	1963-04-06

Logo,

```
1 SELECT COUNT(`Data de Nascimento`) AS `Hóspedes entre 50 e 60 anos`
2 FROM hospede
3 WHERE `Data de Nascimento` BETWEEN '1962-12-31' AND '1972-12-31'
```

Hóspedes entre 50 e 60 anos

5

## Hóspedes + 60 anos

Seguindo a mesma lógica:

```
1 SELECT `Nome Completo`, `Data de Nascimento`
2 FROM hospede
3 WHERE `Data de Nascimento` < '1962-12-31'
```

Resultado:

The screenshot shows a MySQL query results window. At the top, a green bar indicates "MySQL não retornou nenhum registo. (A consulta demorou 0,0000 segundos.)". Below this, the SQL query is displayed: "SELECT `Nome Completo`, `Data de Nascimento` FROM hospede WHERE `Data de Nascimento` < '1962-12-31'". The results table below has two columns: "Nome Completo" and "Data de Nascimento", both of which are empty.

Neste caso, não houve reserva feita por hóspede +60 anos no período de 2022.

Resultado geral:

FAIXA ETÁRIA DOS HÓSPEDES NO ANO DE 2022 NA POUSADA X	
Idade	Nº de hóspedes
10 - 20 anos	10
+ 20 - 30 anos	29
+ 30 - 40 anos	20
+ 40 - 50 anos	9
+ 50 - 60 anos	5
+ 60 anos	0
<b>Nº total de hóspede</b>	<b>73</b>

**Conclusão:** Pode-se constatar que a idade entre +20 e 40 anos são a maioria dos hóspedes, respondendo por, aproximadamente, 67% dos clientes.

## III - Qual Estado é mais recorrente entre os hóspedes?

Para retirada dessa informação, usa-se o comando:

```
1 | SELECT Estado, COUNT(Estado) AS `Nº de hóspede por Estado` FROM hospede GROUP BY Estado
```

Tendo como resultado:

Estado	Nº de hóspede por Estado
BA	5
CE	1
ES	1
GO	1
MA	1
MG	10
PR	6
RJ	28
SC	3
SP	17

**Conclusão:** Logo, sabe-se que a maioria dos clientes que se hospedaram na Pousada X, no ano de 2022, vieram do Estado do Rio de Janeiro (RJ).

#### **IV - Qual foi o mês de mais reservas?**

Para resultar tal informação, fez-se o somatório de reservas para cada mês do ano de 2022. Tendo:

##### **Janeiro**

```
1 | SELECT COUNT(`Check-in`) AS `Nº de reserva para Janeiro`  
2 | FROM `reserva`  
3 | WHERE `Check-in` BETWEEN '2022-01-01' AND '2022-01-31'
```

**Nº de reserva para Janeiro**  
12

##### **Fevereiro**

```
1 | SELECT COUNT(`Check-in`) AS `Nº de reserva para Fevereiro`  
2 | FROM `reserva`  
3 | WHERE `Check-in` BETWEEN '2022-02-01' AND '2022-02-29'
```

**Nº de reserva para Fevereiro**  
9

## Março

```
1 SELECT COUNT(`Check-in`) AS `Nº de reserva para Março`  
2 FROM `reserva`  
3 WHERE `Check-in` BETWEEN '2022-03-01' AND '2022-03-31'
```

Nº de reserva para Março

6

## Abril

```
1 SELECT COUNT(`Check-in`) AS `Nº de reserva para Abril`  
2 FROM `reserva`  
3 WHERE `Check-in` BETWEEN '2022-04-01' AND '2022-04-30'
```

Nº de reserva para Abril

6

## Maio

```
1 SELECT COUNT(`Check-in`) AS `Nº de reserva para Maio`  
2 FROM `reserva`  
3 WHERE `Check-in` BETWEEN '2022-05-01' AND '2022-05-31'
```

Nº de reserva para Maio

5

## Junho

```
1 SELECT COUNT(`Check-in`) AS `Nº de reserva para Junho`  
2 FROM `reserva`  
3 WHERE `Check-in` BETWEEN '2022-06-01' AND '2022-06-30'
```

Nº de reserva para Junho

7

## Julho

```
1 SELECT COUNT(`Check-in`) AS `Nº de reservas para Julho`  
2 FROM `reserva`  
3 WHERE `Check-in` BETWEEN '2022-07-01' AND '2022-07-31'
```

Nº de reservas para Julho

6

## Agosto

```
1 SELECT COUNT(`Check-in`) AS `Nº de reservas para Agosto`  
2 FROM `reserva`  
3 WHERE `Check-in` BETWEEN '2022-08-01' AND '2022-08-30'
```

Nº de reservas para Agosto

5

## Setembro

```
1 SELECT COUNT(`Check-in`) AS `Nº de reservas para Setembro`  
2 FROM `reserva`  
3 WHERE `Check-in` BETWEEN '2022-09-01' AND '2022-09-31'
```

Nº de reservas para Setembro

3

## Outubro

```
1 SELECT COUNT(`Check-in`) AS `Nº de reservas para Outubro`  
2 FROM `reserva`  
3 WHERE `Check-in` BETWEEN '2022-10-01' AND '2022-10-31'
```

Nº de reservas para Outubro

2

## Novembro

```
1 SELECT COUNT(`Check-in`) AS `Nº de reservas para Novembro`  
2 FROM `reserva`  
3 WHERE `Check-in` BETWEEN '2022-11-01' AND '2022-11-30'
```

Nº de reservas para Novembro

6

## Dezembro

```
1 SELECT COUNT(`Check-in`) AS `Nº de reservas para Dezembro`  
2 FROM `reserva`  
3 WHERE `Check-in` BETWEEN '2022-12-01' AND '2022-12-30'
```

Nº de reservas para Dezembro

10

Logo,

RESERVAS DO ANO DE 2022 DIVIDIDA PELOS SEUS RESPECTIVOS MESES	
Mês	Nº de reservas
Janeiro	12
Fevereiro	9
Março	6
Abril	6
Maio	5
Junho	7
Julho	6
Agosto	5
Setembro	3
Outubro	2
Novembro	6
Dezembro	10
<b>Nº total de reservas</b>	<b>77</b>

**Conclusão:** A partir da tabela produzida com as informações anteriores retirada do banco de dados, é notório constatar que os meses de dezembro, janeiro e fevereiro respondem por, aproximadamente, 40% das reservas anual, tendo janeiro como o mês de maior número de reserva.

#### V - Qual o CPF, Nome e Estado do cliente que fez o maior número de reserva em todo ano?

A partir do seguinte comando, tem-se a lista dos clientes que realizaram reservas com o maior número de diárias no ano de 2022.

```
1 SELECT hospede.CPF, hospede.'Nome Completo', hospede.Estado, MAX(reserva.'Numero de diarias') AS 'Nº de reservas'
2 FROM hospede INNER JOIN reserva
3 ON hospede.CPF = reserva.'CPF Hospede'
4 GROUP BY (reserva.'Numero de diarias') DESC
```

Resultado:

CPF	Nome Completo	Estado	Nº de reservas
45685212399	Vanessa Venancio	SP	22
00256497853	Lucas Lottus	BA	16
63985245285	Zemira Zilda	RJ	15
12165678722	Lucas Leite	SP	14
00022266698	Thamyres Tomas	SP	13
01556895522	Alvaro Amancio	GO	9
02589967415	Caio Couto	RJ	8
85274196397	Elias Elpideo	SC	7
00614689566	Gediel Guimaraes	MG	6
00159753864	Marcos Martins	RJ	5
00236985214	Dorgival Dantas	RJ	4
33645217894	Jessica Jurema	RJ	3
79846513200	Henrique Homero	RJ	2

**Conclusão:** Logo, fica-se evidente que a cliente Vanessa Venâncio do Estado de São Paulo do CPF 456.852.123-99 foi a hóspede com a reserva mais longa: 22 diárias.

## VI - Qual a qualificação do hóspede com o maior número de reserva?

A fim de saber qual a qualificação do hóspede de maior número de reserva, foi utilizado o comando “INNER JOIN”, relacionando duas tabelas através da PK CPF.

```
1 SELECT hospede.CPF, hospede.`Nome Completo`, hospede.Estado, reserva.`Qualificacao do Hospede`
2 FROM hospede INNER JOIN reserva
3 ON hospede.CPF = reserva.`CPF Hospede`
4 WHERE hospede.CPF = '45685212399'
```

Resultado:

CPF	Nome Completo	Estado	Qualificacao do Hospede
45685212399	Vanessa Venancio	SP	Bom

**Conclusão:** A hóspede Vanessa Venâncio teve a reserva mais longa e conta com qualificação “bom”.

## 5 - USO DO GIT

Na plataforma Github, foi criado um repositório chamado “projeto-analistasuporte-prion” que receberá os arquivos desenvolvidos no projeto. Para isso, serão exibidos abaixo os comandos GIT que foram usados para o upload dos códigos:

Primeiramente, foi dado o comando “git init”, inicializando um novo repositório.

```
danie@DESKTOP-ABVKJ15 MINGW64 /c/xampp/htdocs/html-css (main)
$ git init
Reinitialized existing Git repository in C:/xampp/htdocs/html-css/.git/
```

Após isso, foi utilizado o “git add .” a fim de adicionar alterações feitas no diretório em atividade. Em seguida, foi dado um “git status” para conferir os arquivos comitados anteriores, porém não registrados.

```
danie@DESKTOP-ABVKJIS MINGW64 /c/xampp/htdocs/html-css (main)
$ git add .

danie@DESKTOP-ABVKJIS MINGW64 /c/xampp/htdocs/html-css (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    deleted:  .gitattributes
    deleted:  LICENSE
    deleted:  README.md
    new file: cad-hospede.css
    new file: cad-hospede.php
    new file: cad-reserva.css
    new file: cad-reserva.php
    new file: conexao-hospede.php
    new file: conexao-reserva.php
    modified: estilo.css
    new file: index.html
    deleted:  pagina_registro_sucesso.html
    deleted:  portfolio_form.html
    new file: processa-hospede.php
    new file: processa-reserva.php
```

O comando “git branch” foi usado para criar uma ramificação, permitindo que futuras alterações possam ser executadas.

```
danie@DESKTOP-ABVKJIS MINGW64 /c/xampp/htdocs/html-css (main)
$ git branch -M main
```

É utilizado, também, o comando “git remote”, com função de interface de gerenciamento, realizando um link direto com o serviço de hospedagem de código, no caso, um repositório do GitHub.

Logo em seguida, o comando “git push” é executado com finalidade de enviar o conteúdo de um repositório local para um repositório remoto, no caso, a hospedagem GitHub.

```
danie@DESKTOP-ABVKJIS MINGW64 /c/xampp/htdocs/html-css (main)
$ git remote add origin https://github.com/daniel19fernandes/projeto-analistasup
orte-prion.git

danie@DESKTOP-ABVKJIS MINGW64 /c/xampp/htdocs/html-css (main)
$ git push -u origin main
Enumerating objects: 19, done.
Counting objects: 100% (19/19), done.
Delta compression using up to 2 threads
Compressing objects: 100% (18/18), done.
Writing objects: 100% (19/19), 7.61 KiB | 1.09 MiB/s, done.
Total 19 (delta 3), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (3/3), done.
To https://github.com/daniel19fernandes/projeto-analistasuporte-prion.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
```

Com tais comandos executados, todos os arquivos foram carregados no repositório “projeto-analista-suporte-prion” no Github, como ilustra a imagem abaixo.

 daniel19fernandes	upload códigos projeto	c95c163 9 minutes ago	 2 commits
 cad-hospede.css	upload códigos projeto	9 minutes ago	
 cad-hospede.php	upload códigos projeto	9 minutes ago	
 cad-reserva.css	upload códigos projeto	9 minutes ago	
 cad-reserva.php	upload códigos projeto	9 minutes ago	
 conexao-hospede.php	upload códigos projeto	9 minutes ago	
 conexao-reserva.php	upload códigos projeto	9 minutes ago	
 estilo.css	upload códigos projeto	9 minutes ago	
 index.html	upload códigos projeto	9 minutes ago	
 processa-hospede.php	upload códigos projeto	9 minutes ago	
 processa-reserva.php	upload códigos projeto	9 minutes ago	

## 6 - CONCLUSÃO

Em linhas gerais, foi apresentado o processo de criação e desenvolvimento do projeto elaborado como complemento e enriquecimento para a vaga de Analista de Suporte, expondo o meu interesse em aprender, desenvolver e ser um colaborador junta ao time Prion.

Ademais, agradeço a oportunidade e gostaria de uma chance para mostrar meu empenho e trabalho.

## 7 - CONTATOS

Daniel Fernandes da Silva

 daniel19fernandes@gmail.com

 [github.com/daniel19fernandes/projeto-analistasuporte-prion](https://github.com/daniel19fernandes/projeto-analistasuporte-prion)

 [linkedin.com/in/daniel19fernandes/](https://linkedin.com/in/daniel19fernandes/)

 (22) 98873-2034