O programa possui como ponto de entrada a função Main. A partir dela, cria-se 5 quartos para exemplificar a execução e pede ao usuário em um laço qual a ação a prosseguir. É nessa etapa que é criada a instância inicial (**Singleton**) do sistema de gerenciamento do hotel.

# 1ª ação – Reservar quarto

O sistema pede o nome do cliente e sua data de nascimento para confirmar se é um cliente já existente. Caso não seja, o sistema pede algumas informações a mais (como endereço e telefone) , e então o cliente é registrado na database. O sistema então pede para o cliente informar o tipo de quarto (simples, tripla, etc) e o tempo de estadia. Após isso, é verificado internamente se há um quarto disponível com essas características. Se houver, a reserva é confirmada pelo cliente. Internamente, o sistema marca o quarto escolhido como “reservado” e dá ao usuário uma identificação única do quarto para o cliente.

# 2ª ação – Listar gastos

Para realizar o controle dos gastos, o sistema itera todos os quartos, imprimindo na tela todos os gastos realizados por cada quarto reservado. Isso é implementado através de um campo na classe QuartoProxy, que marca todos os gastos possíveis de um determinado cliente (telefone, alimentação e diária).

# 3ª ação – Fechar conta

Caso o cliente queira fechar a conta, o sistema pede os identificadores (nome e data de nascimento) para confirmar que a pessoa é realmente um cliente no sistema. Caso confirmado, o quarto reservado pelo cliente é marcado como “disponível”, o cliente é informado os gastos (soma de todas as despesas do quarto) e, após o pagamento, o processo é finalizado. A cobrança é feita por um sistema interno de depósito do gerenciamento do hotel.

# 4ª ação – Relatório diário

Para o relatório do hotel, o sistema realiza uma iteração em todos os quartos reservados e soma os gastos feito pelos quartos, informando cada gasto de cada quarto especificamente. A soma de todos os gastos é a receita do hotel, que aparece na tela do programa no final da função.

# Pattern usado – Singleton

O singleton é usado para representar a instância do sistema de hotel, usado toda vez que o sistema de hotel é acessado no código.

# Pattern usado - Proxy

A classe proxy “QuartoProxy” é usada para representar um quarto no hotel. O quarto pode ser três tipos diferentes: QuartoSimples, QuartoDupla e QuartoTripla. A interface desses três quartos são abstraídas na proxy. O sistema de hotel utiliza o proxy para realizar todas as operações nos quartos.

# Pattern usado – Factory

Factory é usado para gerar uma instância de quarto de um jeito uniforme. O projeto usa três Factory para cada tipo de hotel: QuartoSimplesFactory, QuartoDuplaFactory e QuartoTriplaFactory. São utilizados para criar os quartos exemplo no início da função Main.