**.NET**

**1.** Em um determinado banco, você foi designado para desenhar o novo processamento de liberação de crédito.

Existem 5 tipos de créditos, cujos :

Credito Direto - Taxa de 2% ao mês

Credito Consignado - Taxa de 1% ao mês

Credito Pessoa Jurídica - Taxa de 5% ao mês

Credito Pessoa Física - Taxa de 3% ao mês

Credito Imobiliário - Taxa de 9% ao ano

Cada crédito tem uma forma de validação diferente, porém, todos precisam passar neste método de validação para liberação do crédito.

Implemente as classes que você julgue necessárias para implementação do processamento, e demonstre exemplos em uma aplicação simples de console.

Defina como entradas as seguintes variáveis:

- Valor do crédito

- Tipo de crédito

- Quantidade de parcelas

- Data do primeiro vencimento

As validações das entradas são as seguintes:

- O valor máximo a ser liberado para qualquer tipo de empréstimo é de R$ 1.000.000,00

- A quantidade de parcelas máximas é de 72x e a mínima é de 5x

- Para o crédito de pessoa jurídica, o valor mínimo a ser liberado é de R$ 15.000,00

- A data do primeiro vencimento sempre será no mínimo D+15 (Dia atual + 15 dias), e no máximo, D+40 (Dia atual + 40 dias)

Os resultados precisam conter as seguintes informações :

- Status do crédito (Aprovado ou recusado, de acordo com as premissas acima)

- Valor total com juros

- Valor do juros

Para este exercício, os juros são calculados da seguinte forma, incremente a porcentagem de juros no valor total do crédito.

**Código disponível no meu Github.**

**https://github.com/leonardo-francisco/ TesteLeoFinanciamento**

**ORACLE ou SQL Server**

**2.** Modele uma estrutura de dados do seguinte caso:

- Um cliente tem os seguintes campos : Nome, Id Cliente, UF, Celular

- Um cliente tem N financiamentos.

- Um financiamento tem os seguintes campos : Id Cliente, Tipo Financiamento, Valor Total, Data Vencimento

- Cada financiamento tem N parcelas, cujas tem os seguintes campos : Id Financiamento, Número da Parcela, Valor Parcela, Data Vencimento, Data Pagamento;

Crie as tabelas que julgue necessárias e insira alguns registros de testes na mesma.

Elabore as seguintes querys:

- Listar todos os clientes do estado de SP que tenham mais de 60% das parcelas pagas.

- Listar os primeiros 4 clientes que tenham alguma parcela com mais de 05 dias atrasadas (Data Vencimento maior que data atual E data pagamento nula)

- Listar todos os clientes que já atrasaram em algum momento duas ou mais parcelas em mais de 10 dias, e que o valor do financiamento seja maior que R$ 10.000,00.

**Foi enviado por e-mail o script para geração de tabelas e registros inseridos. Também script para as Querys do questionário.**

**ARQUITETURA**

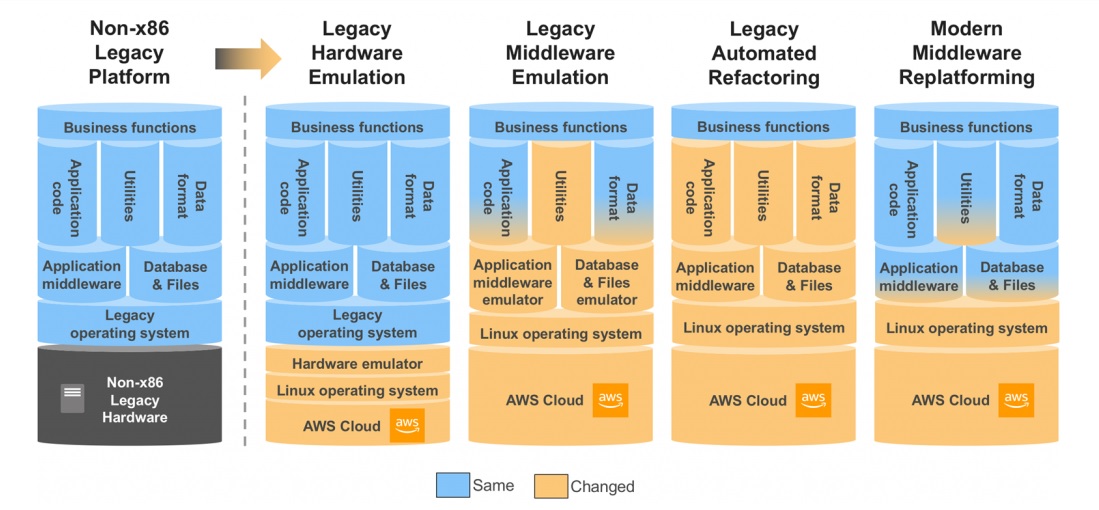
**3.** Questão conceitual, escreva em detalhes ou diagramas como você montaria uma arquitetura para o cenário abaixo, informando as tecnologias, arquitetura, e o que mais achar necessário.

Em tempos de expansão digital, sua empresa foi contratada para desenhar uma arquitetura moderna que sustente o crescimento digital e de vários novos canais, e também tenha formas de manter o legado funcionando.

Descreva ou desenhe o que e como você utilizaria para suportar este crescimento, tendo em vista que é necessário uma arquitetura que agregue os meios de comunicação com Mainframe, e que todos os sistemas web possam se comunicar entre eles sem a reescrita de códigos.

Todos os sistemas são extremamente críticos e de alta performance, também contando com um volume consideravelmente alto de dados sendo transacionados a todo tempo.

Comente também como você desenharia a solução para implantação deste cenário, visando que quanto menor a dependência com áreas de operação para executar a implantação na mão, mais produtivo e assertivo será a empresa.



Escolheria Migração a Curto Prazo. Peguei como inspiração um exemplo que a Amazon usa em seu AWS. Essa opção de migração resulta em uma duração mais curta de projeto para retorno mais rápido sobre o investimento (ROI).

A imagem acima mostra, para cada ação, componentes que mudam e componentes que permanecem os mesmos durante a migração de plataforma.

* **Emulação de hardware Legado**: o emulador de hardware substitui o hardware legado, mas o sistema operacional legado e os aplicativos permanecem os mesmos.
* **Emulação de Middleware Legado**: O emulador de middleware substitui as APIs de middleware legados e as APIs de SO exigidas pelo aplicativo, permitindo a portabilidade. A maior parte do código fonte do aplicativo é recompilada sem alterações, com algumas adaptações para dependências alteradas.
* **Re-fatoração automatizada do legado**: código, dados e dependências são convertidos automaticamente em uma linguagem moderna, armazenamento de dados e frameworks, garantindo equivalência funcional às mesmas funções de negócios.
* **Re-plataforma de Middleware moderno**: aplica-se apenas às linguagens modernas, middleware e runtimes disponíveis em sistemas legados e x86, como Java, PHP e bancos de dados relacionais. Permite reutilizar o código do aplicativo e os bancos de dados.

Crie um repositório público (**Não utilize no projeto ou em qualquer lugar o nome de nenhuma empresa**) e nos envie o link.