

- Pergunta: (Vunesp-2013) Na **depuração** de um **código-fonte**, para **interromper a execução do programa** em determinada **linha** ou **instrução** para **análise**, é necessária...
 - Resposta: A inclusão de **pontos de interrupção (breakpoints)**.
- Pergunta: (Vunesp-2020) Um **programador**, durante o **desenvolvimento** de um **programa**, descobriu que ele não está funcionando de acordo com a **especificação**. No processo de **depuração**, ele resolveu inserir um **breakpoint** nesse **programa**, o que permite...
 - Resposta: **Inspecionar variáveis e conteúdo da memória durante a parada intencional do programa.**
- Pergunta: Qual **componente** abaixo permite **Feedback** de **textos** em uma **aplicação React Native**?
 - Resposta: `<Text>`
- Pergunta: Qual **componente** abaixo permite a **entrada** de **textos** em uma **aplicação React Native**?
 - Resposta: `<TextInput>`
- Pergunta: Assinale qual destas opções corresponde ao **propósito** do **componente ScrollView**.
 - Resposta: O **ScrollView** é um **elemento** com a função de **container**, que pode conter em si vários **elementos**, mantendo uma **altura** previamente **definida** e permitindo a **navegação** entre os **elementos** nele contidos por meio de **rolagem**, que pode ser tanto **vertical** quanto **horizontal**.
- Pergunta: A utilização de **animações** fornece aos **usuários** uma melhor **experiência** no **manuseio** dos **aplicativos mobile**. Em **React Native**, estão disponíveis nativamente duas **APIs** para a **animação** de **elementos**. Falando da **LayoutAnimation API**, é correto afirmar que:
 - Resposta: É uma **API** ainda **experimental**, com algumas **limitações**, sobretudo na **plataforma Android**.
- Pergunta: O **React Native** possui duas **APIs** que permitem a **animação** de **elementos** e **componentes**. A respeito da **Animated API**, é correto afirmar que:
 - Resposta: A **Animated API** permite a criação de **animações configuráveis** por meio de **instruções**. Aliás, é possível criar **componentes personalizados** com o método **Animated.CreateAnimatedComponent()**.
- Pergunta: (FCC/2019) Um **Analista** precisa **desenvolver** um **aplicativo móvel** para **celulares** com sistemas operacionais **Android** e **iOS**. Para isso, poderá utilizar o **framework** desenvolvido pela equipe do **Facebook**, que possibilita o **desenvolvimento** de **aplicações mobile** utilizando **bibliotecas JavaScript** para criar **interfaces de usuário**. Esse **framework** é conhecido como:
 - Resposta: **React Native**.
- Pergunta: Tanto **Android** quanto **iOS** apresentam **estruturas** para **armazenamento seguro** de **dados simples**, normalmente associados à **configuração** dos **aplicativos**, como **Keychain**, para o **iOS**, e **EncryptedShared Preferences**, no ambiente **Android**. Qual **componente** deve ser utilizado, no **desenvolvimento** com **React Native**, para **encapsular** os **mecanismos básicos** de **gravação segura** de ambas as **plataformas**?
 - Resposta: **EncryptedStorage**

- Pergunta: O ambiente do **React Native** oferece **componentes** para as mais diversas situações, podendo contar com algumas **bibliotecas nativas** e outras **importadas** via **NPM** ou **VARN**. Qual **biblioteca** deve ser **importada** para permitir a **escrita e leitura de pares chave-valor em arquivos de texto**?
- Resposta: @react-native-async-storage/async-storage
- Pergunta: As diversas áreas do **desenvolvimento de software** sempre estão associadas a diversos **ferramentais da matemática**. No caso específico dos **bancos de dados relacionais**, quais são os **elementos matemáticos** que os regem?
- Resposta: **Lógica Booleana e Álgebra Relacional**.
- Pergunta: Os **bancos de dados orientados a objetos** permitem que utilizemos um **sistema de armazenamento** com **estrutura** similar à utilizada pelas **linguagens de programação** da atualidade, diminuindo a **distância semântica** do **modelo relacional**. Qual alternativa se refere a um **banco de dados** que, embora também use o **modelo documental**, é reconhecido como **orientado a objetos**?
- Resposta: **Realm**
- Pergunta: Considere um **aplicativo React Native** para **controle de entregas**, com uma **base SQLite** que é alimentada a partir do **servidor** diariamente, criado na **arquitetura MVC**, trabalhando com as **entidades Cliente e Endereço**, ambas já definidas ao nível da **camada Model**, bem como respectivos **componentes DAO**. Caso seja necessária uma **busca de CLIENTES por BAIRRO**, do tipo **texto**, como seria a **assinatura do método** ao nível do **CONTROLLER**?
- Resposta: `public Cliente[] obterBairro(String bairro) {/**/}`
- Pergunta: Em termos de **implementação** de um **aplicativo** que faça uso do **modelo Offline First** temos, como **elemento**, o **banco de dados embarcado**. Sobre tal **recurso** e seu uso nesse **modelo**, é correto afirmar:
- Resposta: São **indispensáveis**, uma vez que precisaremos ter um **mecanismo** que possibilite a **persistência e recuperação local** dos **dados** necessários ao **funcionamento** do **aplicativo** para **posterior sincronização**.
- Pergunta: A respeito do **OAuth2** é correto afirmar:
- Resposta: É um **framework** de **autorização** que permite a um **aplicativo** ter **acesso limitado e controlado** a **recursos de terceiros**, como **APIs**.
- Pergunta: Um **componente Reducer** é parte integrante da **arquitetura Redux**, e tem como objetivo efetuar os **processos** necessários para a **mudança do estado** representado pelo **componente Store**. Quais são os **requisitos** para a **definição** de uma **função Reducer**?
- Resposta: Ela deve ter como **entrada** um **estado inicial** e uma **ação**, além de **retornar o estado modificado** ao final da **execução**.