

## As partes da API GraphQL

Depois de definirmos alguns tipos e criarmos queries no GraphQL, já temos uma ideia melhor sobre como as partes principais da API trabalham juntas. Então podemos afirmar que:

Selecione as alternativas corretas:

*Selecione 3 alternativas*

- A** O argumento `context` é um objeto que a função `resolver` recebe com informações e dados para resolver a query — por exemplo, como acessar os dados.
- B** Todos os campos e subcampos de um tipo GraphQL devem obrigatoriamente estar “associados” a uma função `resolver`.
- C** No projeto que estamos implementando, a camada `dataSource` é responsável por se conectar aos endpoints REST através de seus métodos, como `.get()`.
- D** Os tipos que indicam os possíveis pontos de entrada de uma API GraphQL são *Query Types* ou também chamados de *root types*.