

Adicionando conversor de JSON completo

Dentro do método `doInBackground()` da `InsereAlunoTask`, inicialmente, faremos a chamada do responsável em converter o objeto aluno para JSON. Portanto, faça a chamada da classe `AlunoConverter`:

```
@Override  
protected Object doInBackground(Object[] params) {  
    new AlunoConverter();  
    return null;  
}
```

Agora que chamamos o conversor, precisamos criar um método que ficará responsável em realizar a conversão completa do objeto aluno. Em outras palavras, adicione o trecho `.converteParaJsonCompleto(aluno)` (enviando o aluno que será convertido! :)) na chamada da classe `AlunoConverter` e atribua essa chamada de método para uma variável do tipo `String` com o nome `json`, conforme o código abaixo:

```
@Override  
protected Object doInBackground(Object[] params) {  
    String json = new AlunoConverter().converteParaJsonCompleto(aluno);  
    return null;  
}
```

Repara que o Android Studio estará reclamando, pois não conhece o método `converteParaJsonCompleto(aluno)`. Portanto, vamos criá-lo! Faremos isso da mesma forma como fizemos com a `InsereAlunoTask`, ou seja, move o cursor do mouse em cima do método e pressione **Alt + Enter** e escolha a opção **Create method** '`converteParaJsonCompleto`'. O Android Studio criará o método dentro da classe `AlunoConverter` com a seguinte assinatura e corpo:

```
public String converteParaJsonCompleto(Aluno aluno) {  
    return null;  
}
```

Em seguida, adicione o código que fará essa conversão para nós! Ou seja, aquele código que faz uso da classe `JSONStringer` e que vai adicionando uma chave (`key()`) e o valor (`value()`) para cada atributo da classe `Aluno`. Lembrando que para criarmos um objeto a partir da classe `JSONStringer`, precisamos chamar o método `object()`.

