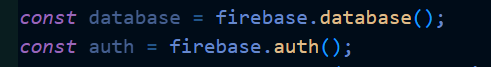
**Conceitos Básicos**

**Instancionou objetos**

* Quais objetos?

R: Foram instanciados os objetos database e auth das classes fornecidas pelo Firebase SDK



**código completo:** <https://github.com/ProjetoIntegrador23/projetoIntegrador23/blob/main/js/pages/historic.js>

**Codificou atributos protegidos e/ou privados (**

* Em quais classes? Quais atributos?

**R:** Esses atributos privados foram declarados na classe SensorACS712 e eles são os seguintes:

**class SensorACS712 {**

**public:**

**SensorACS712(int pino, float voltsPorUnidade, float sensibilidade);**

**void lerSensor();**

**void imprimirDados();**

**private:**

**int pinoSensor;**

**float voltsPorUnidade;**

**float sensibilidade;**

**float corrente;**

**float tensao;**

**float potencia;**

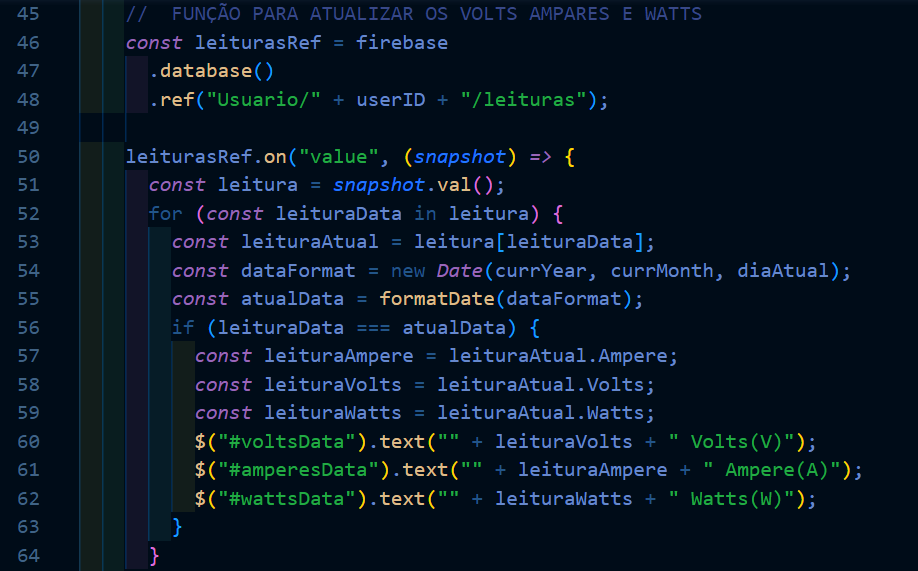
**};**

* Por que não são públicos?

**R:** Eles são privados, pois se tratam das informações geradas pelo sensor de corrente e é crucial que nenhum usuário a modifiquem, um exemplo disso seria a sensibilidade na qual seu valor é definida de acordo com relação entre a corrente medida e a tensão de saída do sensor, ou seja, é um valor que não pode ser alterado.

**código completo:** https://github.com/ProjetoIntegrador23/projetoIntegrador23/blob/main/Circuito/VERSAO\_FINAL\_PROJETO.txt

**Codificou propriedades**Quais propriedades?  
R: O código está usando propriedade de objetos, por exemplo, o objeto de leitura. Nessa parte do código é atualizado a leitura do circuito em volts, amperes e watts.



**Design - 10 pontos**

**Identificou e codificou classes de dados**

* Quais classes?

A partir do método lerSensor da classe SensorACS712 foi possível fazer a leitura e tratar os dados obtidos pelo sensor por meio da manipulação dos atributos declarados na classe.

| **void SensorACS712::lerSensor() {**  **int valorSensor = analogRead(pinoSensor);**  **corrente = (valorSensor \* voltsPorUnidade) / sensibilidade;**  **tensao = valorSensor \* voltsPorUnidade;**  **potencia = corrente \* tensao;**  **}** |
| --- |

**código completo:** https://github.com/ProjetoIntegrador23/projetoIntegrador23/blob/main/Circuito/VERSAO\_FINAL\_PROJETO.txt

**Identificou e codificou classes de comportamento**

* Quais classes?

R: Na SensorACS712 é usado os métodos lerSensor e imprimirDados que tem comportamentos específicos, garantindo a organização do código na hora de realizar os cálculos desejados e imprimir os resultados.

| **class SensorACS712 {**  **public:**  **SensorACS712(int pino, float voltsPorUnidade, float sensibilidade);**  **void lerSensor();**  **void imprimirDados();**  **private:**  **int pinoSensor;**  **float voltsPorUnidade;**  **float sensibilidade;**  **float corrente;**  **float tensao;**  **float potencia;**  **};**  **void SensorACS712::lerSensor() {**  **int valorSensor = analogRead(pinoSensor);**  **corrente = (valorSensor \* voltsPorUnidade) / sensibilidade;**  **tensao = valorSensor \* voltsPorUnidade;**  **potencia = corrente \* tensao;**  **}** |
| --- |

**código completo:** https://github.com/ProjetoIntegrador23/projetoIntegrador23/blob/main/Circuito/VERSAO\_FINAL\_PROJETO.txt

**Ocultou informações usando atributos e ou métodos protected/private 1 2:**

* Em quais classes?

R: Na classe SensorACS712

* Quais atributos e métodos?

R: Os atributos declarados como privados foram pinoSensor, voltsPorUnidade, sensibilidade, corrente, tensao e potencia.

* Por que foi importante ocultar esses dados?

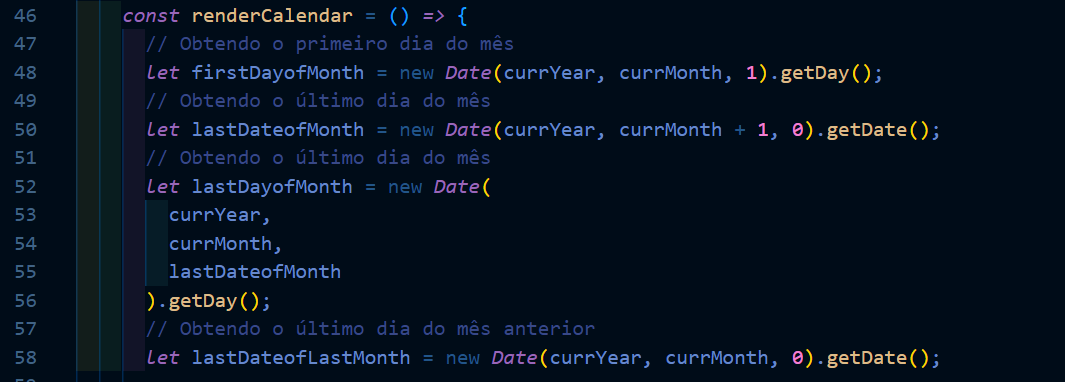
R: Para que esses dados sejam acessados apenas dentro da própria classe e nenhum usuário externo tenha acesso a elas, já que se trata de valores que não podem ser modificados por outra pessoa.

| **class SensorACS712 {**  **public:**  **SensorACS712(int pino, float voltsPorUnidade, float sensibilidade);**  **void lerSensor();**  **void imprimirDados();**  **private:**  **int pinoSensor;**  **float voltsPorUnidade;**  **float sensibilidade;**  **float corrente;**  **float tensao;**  **float potencia;**  **};**  **void SensorACS712::lerSensor() {**  **int valorSensor = analogRead(pinoSensor);**  **corrente = (valorSensor \* voltsPorUnidade) / sensibilidade;**  **tensao = valorSensor \* voltsPorUnidade;**  **potencia = corrente \* tensao;**  **}** |
| --- |

**código completo:** https://github.com/ProjetoIntegrador23/projetoIntegrador23/blob/main/Circuito/VERSAO\_FINAL\_PROJETO.txt

**Boas práticas**

**Usou conceitos de código limpo**

Quais conceitos? Onde?  
R:No código foi usado algumas práticas de organização e legibilidade. Na programação do calendário em nosso site a função e as variáveis têm os nomes bem objetivo conforme a sua funcionalidade.   
 **código completo:** <https://github.com/ProjetoIntegrador23/projetoIntegrador23/blob/main/js/pages/historic.js>