

1) Array literal, Literais boolean, Literais de ponto flutuante, Inteiros, Objeto literal, String literal.

2) application/json

3) O valor recebido por "null" na linha 03 deveria ser null, ao invés de nil

```
4) {      "cadence":setCadence,
          "gear": setGear,
          "speed": [applyBrake, speedUp]
      }
```

```
5) {      "cidade":{Goiânia, trindade},
          "salada": "alface",
          "altura": 2, 3,
          "sabor" : {}
      }
```

6) Ok!

7) package aula;

```
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.FileReader;
import java.io.IOException;
import java.io.InputStream;
import java.net.URL;
import java.net.URLConnection;

import org.json.simple.JSONObject;
import org.json.simple.parser.JSONParser;
import org.json.simple.parser.ParseException;

public class Aula {

    public static void main(String[] args) throws IOException {

        // Obtem arquivo JSON da API do GitHub
        URL url = new URL("https://api.github.com/users/rhenanrk");
        URLConnection connection = url.openConnection();
        InputStream in = connection.getInputStream();
        FileOutputStream fos = new FileOutputStream(new File("idGit.json"));
        byte[] buf = new byte[512];
        while (true) {
            int len = in.read(buf);
            if (len == -1) {
                break;
            }
            fos.write(buf, 0, len);
        }
        in.close();
        fos.flush();
    }
}
```

```

        fos.close();

        // Faz o parser e ontem o URL do avatar
        JSONObject jsonObject; //Cria o parse de tratamento
        JSONParser parser = new JSONParser(); //Variaveis que irao armazenar os dados do arquivo JSON
        String avatar_url;

        try {
            //Salva no objeto JSONObject o que o parse tratou do arquivo
            jsonObject = (JSONObject) parser.parse(new FileReader("idGit.json"));

            //Salva nas variaveis os dados retirados do arquivo
            avatar_url = (String) jsonObject.get("avatar_url");

            System.out.printf("Nome: %s\n", avatar_url);
        } //Trata as exceptions que podem ser lançadas no decorrer do processo
        catch (FileNotFoundException e) {
            e.printStackTrace();
        } catch (IOException e) {
            e.printStackTrace();
        } catch (ParseException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }

        // Obtem o arquivo de imagem a partir da URL do avatar
        url = new URL("https://avatars2.githubusercontent.com/u/16082661?v=4");
        connection = url.openConnection();
        in = connection.getInputStream();
        fos = new FileOutputStream(new File("idGit.jpeg"));
        buf = new byte[512];

        while (true) {
            int len = in.read(buf);
            if (len == -1) {
                break;
            }
            fos.write(buf, 0, len);
        }

        in.close();
        fos.flush();
        fos.close();
    }
}

```