

PROJETO RELATÓRIO ILHAS POR LOTE

O objetivo do programa é rodar de forma mais rápida todo o processo.

O sistema é dividido em 8 Processos:

- [Processo MATERIALIZEILHAS](#MATERIALIZEILHAS)

- [Processo SLEDGEHAMMER](#SLEDGEHAMMER)
- [Processo DETALHADO](#DETALHADO)
- [Processo PEDAGOGICO](#PEDAGOGICO)
- [Processo LETRAS](#LETRAS)
- [Processo FONEMA](#FONEMA)
- [Processo CALIGRAFIA](#CALIGRAFIA)
- Entendimento do Código Java](#ENTENDIMENTO)
- [Processo EMAIL](#EMAIL)

Processo MATERIALIZEILHAS

Processo MATERIALIZEILHAS

Esse processo consiste em criar uma view materializada, para que a consulta seja mais rápida.

Para que os 7 processos abaixo possam ter êxito é **NECESSÁRIO** que esse processo seja executado na madrugada.

O processo precisa sempre dar um **REFRESH** para que possa pegar a data atual. O código para criar as views estão no banco de dados, e para que possa dar refresh basta colocar no banco o seguinte script:

Esse **Processo de Materialize de views** Foi substituído por um processo mais eficiente e rápido. Ver a documentação do processo de Materialize.

SLEDGEHAMMER

• Processo SLEDGEHAMMER

Esse processo consistem em corrigir as informações da tabela **ILHAS.USERLOG**. Alguns usuários estão passando o tempo jogado 1 hora, muitas vezes deixando o tablet em standby.

Exemplo:

```
[USERLOGID](/cgi-bin/view/Main/USERLOGID) :910324
[USERID](/cgi-bin/view/Main/USERID) :27635
[STARTDATETIME](/cgi-bin/view/Main/STARTDATETIME) :2018-
11-12 17:12:47
[FINISHDATETIME](/cgi-bin/view/Main/FINISHDATETIME)-[ANTERIOR](/cgi-
bin/view/Main/ANTERIOR) :2018-11-13 10:53:37
[FINISHDATETIME](/cgi-bin/view/Main/FINISHDATETIME)-[CORRIGIDO]
(/cgi-bin/view/Main/CORRIGIDO) :2018-11-12 17:17:47
[TEMPO](/cgi-bin/view/Main/TEMPO) [ANTERIOR](/cgi-
bin/view/Main/ANTERIOR) :0 [DIAS](/cgi-bin/view/Main/DIAS) <
```

```
= > 17h : 40m : 50s  
[TEMPO](/cgi-bin/view/Main/TEMPO) [CORRIGIDO](/cgi-  
bin/view/Main/CORRIGIDO) :00:05:00
```

Como pode ver o **STARTDATETIME** está com a data **2018-11-12 17:12:47** e o **FINISHDATETIME-ANTERIOR** esta com um dia á mais, ou seja o aluno , não jogaria dois dias direto.

Ao rodar esse processo, ele envia o id da tabela e o id do aluno para poder conferir as informação se necessário.

DETALHADO

- **Processo DETALHADO**

Esse processo é o principal.

É o gerencial, onde contem todas as informações dos alunos, classes, turmas e escolas.

PEDAGOGICO

- **Processo PEDAGOGICO**

Esse processo pega os dados dos jogos pedagógicos.

LETRAS

- **Processo LETRAS**

Esse processo pega os dados dos jogos Letras.

FONEMA

- **Processo FONEMA**

Esse processo pega os dados dos jogos Fonema.

CALIGRAFIA

- **Processo CALIGRAFIA**

Esse processo pega os dados dos jogos Caligrafia

ENTENDIMENTO

- **Entendimento**

Aqui vou colocar um breve entendimento do código.

O projeto se encontra no [GIT](#) versionado com o nome [ReportIlhasLote](#).

O projeto consistem em 5 packages:

- connectionfactory
- controller
- model

- sendmail
- view

A package **connectionfactory** contém as classes de conexão com o banco de dados, seja Homologação, Desenvolvimento ou Produção.

Aqui irei colocar os exemplos de classes de conexão.

```
package connectionfactory

import java.sql.Connection;
import java.sql.[DriverManager](/cgi-bin/view/Main/DriverManager);

public class [ConnectionFactoryDev](/cgi-bin/view/Main/ConnectionFactoryDev) {

    String string[DeConexao](/cgi-bin/view/Main/DeConexao) =
    "jdbc:postgresql://localhost:5432/abs"; //url do driver jdbc
    String usuario = "postgres"; //seu usuario do banco de dados
    String senha = "idades"; //sua senha do banco de dados
    Connection conexao = null;

    try{
        conexao = [DriverManager](/cgi-bin/view/Main/DriverManager).getConnection(string[DeConexao](/cgi-bin/view/Main/DeConexao), usuario, senha); //cria uma conexão
    } catch(Exception e ){
        e.print[StackTrace](/cgi-bin/view/Main/StackTrace)();
    }
    return conexao; //retorna a conexão
}
```

Essa classe abaixo é que solicitamos a conexão com o banco

```
package connectionfactory;

import java.sql.Connection;
import java.sql.[SQLException](/cgi-bin/view/Main/SQLException);

public class Conexao {
    public Connection connection;

    public Conexao() {

        connection = new [ConnectionFactoryProd](/cgi-bin/view/Main/ConnectionFactoryProd)().createConnection(); // [PRODUCAO] (/cgi-bin/view/Main/PRODUCAO)
        // connection = new [ConnectionFactoryDev](/cgi-bin/view/Main/ConnectionFactoryDev)().createConnection(); //[DEV](/cgi-bin/view/Main/DEV)
```

```

    }

    public void close() throws [SQLException](/cgi-bin/view/Main/SQLException) {
        // [TODO](/cgi-bin/view/Main/TODO) Auto-generated method stub
        connection.close();
    }
}

```

Para que possamos chamar a conexão com o banco, basta colocar nas classes o seguinte código.

```
Conexao conProd = new Conexao();
```

A package **CONTROLLER** fica todas as classes de regras e execução dos processos

- Roda o Processo sledgehammer
 - Roda o Processo detalhado
 - Roda o Processo pedagógico
 - Roda o Processo letras
 - Roda o Processo fonema
 - Roda o Processo caligrafia
 - Roda o Processo [WriteTable](#) #Faz o insert na tabela **report.tb_check_log_ilhas**
 - Roda o Processo [WriteReport](#) #Gera o arquivo de log **LogRelatorioIlhasdoAlfaeBeto.txt** esse arquivo é gravado na tabela **"/tmp"**
 - Roda o Process [WriteLog](#) #Esse processo consiste em gerar os arquivos **".sql"**

A package **MODEL** fica todas as classes de **ENUM**.

Dentro dessa package existe duas classes: - [TabRelatorioExec](#) - [SeqTabelasRelatExec](#)

Essas classes tem o nome de todas as tabelas e sequências do projeto, caso haja alteração do nome , basta apenas trocar o nome nessas classes, sem a necessidade de trocar em todas as classes ou scripts do projeto.

A package **SENDMAIL** fica a classe para envio de email.

Foi criado um email para utilização de envio de relatórios:

usuário: logrelatoriosiab@gmail.com

senha: l@b2018#

Ela consegue pegar o arquivo em anexo e enviar para os e-mails já estabelecidos, que são:

- diego@idados.org.br
- daniel@idados.org.br

- joas@alfaebeto.org.br
- ludimila@alfaebeto.org.br
- renato@alfaebeto.org.br
- maicon@idados.org.br