**Controller:**

ApiCalls:

**package** com.example.newsapp2.controller;

**import** java.io.IOException;

**import** java.net.URI;

**import** java.net.http.HttpClient;

**import** java.net.http.HttpRequest;

**import** java.net.http.HttpResponse;

**import** java.sql.Date;

**import** java.text.ParseException;

**import** java.text.SimpleDateFormat;

**public** **class** ApiCalls {

//API Call to get leagues of the different games

**public** String newsGlance() **throws** IOException, InterruptedException

{

HttpRequest request = HttpRequest.newBuilder()

.uri(URI.create("https://bing-news-search1.p.rapidapi.com/news/trendingtopics?cc=IN&textFormat=Raw&safeSearch=Off"))

.header("x-bingapis-sdk", "true")

.header("x-rapidapi-key", "4e96ca0c82msh7acc5424070fda1p1c7d82jsn9ec25a33485e")

.header("x-rapidapi-host", "bing-news-search1.p.rapidapi.com")

.method("GET", HttpRequest.BodyPublishers.noBody())

.build();

HttpResponse<String> response = HttpClient.newHttpClient().send(request, HttpResponse.BodyHandlers.ofString());

System.out.println(response.body());

**return** response.body();

}

//API Call to get News of the different games

**public** String news() **throws** IOException, InterruptedException

{

HttpRequest request = HttpRequest.newBuilder()

.uri(URI.create("https://bing-news-search1.p.rapidapi.com/news?safeSearch=Off&textFormat=Raw"))

.header("x-bingapis-sdk", "true")

.header("x-rapidapi-key", "4e96ca0c82msh7acc5424070fda1p1c7d82jsn9ec25a33485e")

.header("x-rapidapi-host", "bing-news-search1.p.rapidapi.com")

.method("GET", HttpRequest.BodyPublishers.noBody())

.build();

HttpResponse<String> response = HttpClient.newHttpClient().send(request, HttpResponse.BodyHandlers.ofString());

System.out.println(response.body());

**return** response.body();

}

//API Call to get live matches of the different games

**public** String srch(String str) **throws** IOException, InterruptedException

{

HttpRequest request = HttpRequest.newBuilder()

.uri(URI.create("https://bing-news-search1.p.rapidapi.com/news/search?q=inbody%3A"+str+"%20OR%20intitle%3A"+str+"&freshness=Day&textFormat=Raw&safeSearch=Off"))

.header("x-bingapis-sdk", "true")

.header("x-rapidapi-key", "4e96ca0c82msh7acc5424070fda1p1c7d82jsn9ec25a33485e")

.header("x-rapidapi-host", "bing-news-search1.p.rapidapi.com")

.method("GET", HttpRequest.BodyPublishers.noBody())

.build();

HttpResponse<String> response = HttpClient.newHttpClient().send(request, HttpResponse.BodyHandlers.ofString());

System.out.println(response.body());

**return** response.body();

}

}

CONTROLLER:

**package** com.example.newsapp2.controller;

**import** java.io.IOException;

**import** java.util.ArrayList;

**import** java.util.Iterator;

**import** java.util.Map;

**import** java.util.Spliterators;

**import** java.util.stream.Collectors;

**import** java.util.stream.StreamSupport;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.stereotype.Controller;

**import** org.springframework.web.bind.annotation.RequestMapping;

**import** org.springframework.web.bind.annotation.RequestMethod;

**import** org.springframework.web.servlet.ModelAndView;

**import** com.example.newsapp2.controller.ApiCalls;

**import** com.example.newsapp2.controller.Input;

**import** com.example.newsapp2.controller.Repo;

**import** com.fasterxml.jackson.databind.ObjectMapper;

@Controller

**public** **class** controller {

ApiCalls ap=**new** ApiCalls(); //create object of the ApiCall.java class to access methods inside it

@Autowired

Repo repo;

@RequestMapping(value="/", method=RequestMethod.***GET***) //it will display index.html file to the user whenever "/" request is made.

**public** ModelAndView home(ModelAndView modelAndView)

{

modelAndView.setViewName("index");

**return** modelAndView;

}

@RequestMapping(value="/ng", method=RequestMethod.***GET***) //it will display index.html file to the user whenever "/" request is made.

**public** ModelAndView nghome(ModelAndView modelAndView) **throws** IOException, InterruptedException

{

String map=ap.newsGlance();

ObjectMapper mapper = **new** ObjectMapper();

Map<String, Object> RecMap=mapper.readValue(map, Map.**class**);

ArrayList<String> data=(ArrayList<String>)RecMap.get("value");

// System.out.println("data is: "+data);

Iterator it=data.iterator();

Iterable<Object> iterable = (Iterable<Object>) StreamSupport.*stream*(Spliterators.*spliteratorUnknownSize*(it, 0),**false**).collect(Collectors.*toList*());

modelAndView.addObject("newsList",iterable);

modelAndView.setViewName("news");

**return** modelAndView;

}

@RequestMapping(value="/news", method=RequestMethod.***GET***) //it will display index.html file to the user whenever "/" request is made.

**public** ModelAndView nhome(ModelAndView modelAndView) **throws** IOException, InterruptedException

{

String map=ap.news();

ObjectMapper mapper = **new** ObjectMapper();

Map<String, Object> RecMap=mapper.readValue(map, Map.**class**);

ArrayList<String> data=(ArrayList<String>)RecMap.get("value");

// System.out.println("data is: "+data);

Iterator it=data.iterator();

Iterable<Object> iterable = (Iterable<Object>) StreamSupport.*stream*(Spliterators.*spliteratorUnknownSize*(it, 0),**false**).collect(Collectors.*toList*());

modelAndView.addObject("newsList",iterable);

modelAndView.setViewName("news2");

**return** modelAndView;

}

@RequestMapping(value="/srch", method=RequestMethod.***GET***) //it will display index.html file to the user whenever "/" request is made.

**public** ModelAndView srgethome(ModelAndView modelAndView,Input input) **throws** IOException, InterruptedException

{

modelAndView.addObject("input",input);

modelAndView.setViewName("search");

**return** modelAndView;

}

@RequestMapping(value="/srch", method=RequestMethod.***POST***) //it will display index.html file to the user whenever "/" request is made.

**public** ModelAndView srhome(ModelAndView modelAndView,Input input) **throws** IOException, InterruptedException

{

String map=ap.srch(input.getWord());

ObjectMapper mapper = **new** ObjectMapper();

Map<String, Object> RecMap=mapper.readValue(map, Map.**class**);

ArrayList<String> data=(ArrayList<String>)RecMap.get("value");

System.***out***.println("data is: "+data);

Iterator it=data.iterator();

Iterable<Object> iterable = (Iterable<Object>) StreamSupport.*stream*(Spliterators.*spliteratorUnknownSize*(it, 0),**false**).collect(Collectors.*toList*());

modelAndView.addObject("input",input);

modelAndView.addObject("news",iterable);

modelAndView.setViewName("search");

**return** modelAndView;

}

}