



package some.job.description;

import com.daimler.tss.TeamExpectations;

/**

*
* @author Daimler TSS GmbH
*
* Das Ziel unseres Fachgebiets 'System Integration Sales' ist es, ein herausragendes Endkundenerlebnis im Umfeld Mercedes-Benz-Retail zu gestalten.

* Hierfür betreuen wir ein breites Applikationsportfolio: vom europäischen MB Retail Verkaufsportal,
* über IAM Lösungen und einer Datenaustauschplattform, bis zu fahrzeugnahen ConnectMe Aktivierungsdiensten.

*
* Unser Team besteht aus erfahrenen Spezialisten aus den Themenfeldern SW-Entwicklung, IT-Betrieb, Integration und
* Retail Sales Prozesse. We build it, we run it!

*
* 2018 wurde uns die Auszeichnung „Great Place to Work – Beste Arbeitgeber 2018“ verliehen.
* Also warum sollten wir nicht auch Dein Great Place to Work sein?

*/

public class JobDescription {

/**

* Wir suchen leidenschaftliche SW-Entwickler, die Spaß daran haben die Digitalisierung des Mercedes Benz Vertriebs voranzubringen

*/

List<String> workDescription= Arrays.asList("Du entwickelst Java Anwendungen mit Fokus auf Backend und Schnittstellen",
"Du arbeitest gerne mit einem hoch motivierten Team zusammen und tauscht Dich gerne im Team aus",
"Du konzipierst Design- & Lösungsvorschläge und setzt diese um");

static class Employer {

static List<String> requirements = Arrays.asList("Abgeschlossenes Studium der Informatik / vergleichbarer Studiengang, eine Ausbildung zum Fachinformatiker mit Schwerpunkt SW-Entwicklung oder entsprechende praktische Erfahrung",

"Erfahrung in der Entwicklung von Java Backend Applikationen mit modernen Frameworks (Spring, REST, ect.) in verteilten Architekturen",

"Erfahrung in der Entwicklung mit Datenbanken (relationale und/oder NoSQL)",

"Leidenschaft für die Entwicklung von Software und die stetige Verbesserung der SW-Qualität ('Clean Code') auch durch eine hohe Testabdeckung ('TDD')",

"Neue Technologien und Methoden findest du spannend und suchst, wo sinnvoll, Einsatzgebiete, die einen Mehrwert für unsere Anwender bringen",

"Erfahrung mit Tools zur Versionskontrolle",

"Build-Werkzeuge & Testautomatisierung wünschenswert",

"Erfahrung mit agiler SW-Entwicklung und DevOps wünschenswert"

);

Map<String, List<String>> technicalStack = initTechMap();

private Map<String, List<String>> initTechMap(){
Map<String, List<String>> tech = **new** HashMap<>();
tech.put("Backend",
Arrays.asList("Java 8", "Spring-Boot", "Java EE"));
tech.put("CI/CD",
Arrays.asList("Maven", "GIT", "Nexus", "Jenkins", "SonarCube", "Checkmarx", "BlackDuck", "ELK/Splunk", "LoadRunner/JMeter", "AppDynamics"));
tech.put("Runtime Platform",
Arrays.asList("Kubernetes", "Docker"));
tech.put("Database",
Arrays.asList("MariaDB Galera Cluster"));
tech.put("Web Frontend",
Arrays.asList("Angular Framework / Typescript"));
return tech;
}

private List<String> benefits = Arrays.asList("Kollegen mit denen man Spaß haben kann",
"Chill-Ecken mit Spielekonsolen",
"30 Urlaubstage, flexible Arbeitszeiten, mobiles Arbeiten, Freizeitausgleich für Überstunden, Sonderurlaub für besondere Ereignisse sowie Sabbatical",
"Modern ausgestattete Arbeitsplätze, Nutzung neuester Technologien, umfassendes Angebot an internen und externen Weiterbildungsmaßnahmen sowie Knowledge Upgrade Lectures",
"Wasserspender, kostenloser Kaffee & Tee, kostenlose Obstkörbe und subventionierte Essensangebote",
"Betriebliche Altersvorsorge, Betriebskrankenkasse der Daimler AG und Abschluss einer privaten Unfallversicherung",
"Firmenangehörigengeschäft der Daimler AG, und Bezuschussung des VVS Tickets in der Region Stuttgart und Ulm",
"TSS Mitarbeiterevent sowie Weihnachtsfeier"

);

class Employee {

boolean understandsStreamAndLambda;

boolean likesTechnicalExploration;

int skillLevel = 0;

public int validateSkills(List<String> skillset) {

skillLevel = skillset.stream()

.filter(e -> Employer.requirements.contains(e))

.collect(Collectors.toList())

.size();

if(understandsStreamAndLambda && likesTechnicalExploration)
skillLevel++;

return skillLevel;

}

public String advicelfAplyOrNot() {

if(skillLevel >= TeamExpectations.PRO_DEVELOPER)

return "WT*?! Bewerben!";

else

return "No risk, no success ;)";

}

}

//TODO: If it smells: Refactor and improve

}