

```
//SPDX-License-Identifier:MIT
```

```
pragma solidity >=0.8.12;
```

```
contract JobPortal{
```

```
    address public Admin;
```

```
    constructor(){
```

```
        Admin=msg.sender;
```

```
    }
```

```
    modifier onlyAdmin() {
```

```
        require(msg.sender==Admin, "Unauthorized access to the portal");
```

```
        _;
```

```
    }
```

```
    struct Applicant{
```

```
        string name; string location; string applicanttype;
```

```
        uint8 age; uint8 rating;
```

```
        uint8 applicantID;
```

```
    }
```

```
    Applicant private applicant;
```

```
    Applicant[] private applicant_details;
```

```
    struct Job{
```

```
        uint8 jobid;
```

```
        string jobtitle; string location;
```

```
        uint256 remuneration;
```

```

}

Job private job;

Job[] private job_details;


// function to add a new applicant (only admin can do it).


function add_new_applicant(string memory _name, string memory _location, string memory
_applicanttype, uint8 _age, uint8 _Id) public onlyAdmin{

    applicant.name=_name;

    applicant.location=_location;

    applicant.applicanttype=_applicanttype;

    applicant.age=_age;

    applicant.applicantID=_Id;

    applicant_details.push(applicant); }


//function to get an applicant's details

function get_applicant_details(uint8 applID) public view returns(Applicant memory){

    return(applicant_details[applID]);

}


//function to get the applicant type

function fetch_applicanttype(uint8 _ID) public view returns(string memory){

    return(applicant_details[_ID].applicanttype);

}


// function to add a new job, only admin can do it

function add_new_job(uint8 _jobid, string memory _jobtitle, string memory _location, uint256
_remuneration) public onlyAdmin{

```

```

    job.jobid=_jobid;
    job.jobtitle=_jobtitle;
    job.location=_location;
    job.remuneration=_remuneration;
    job_details.push(job);

}

//function to get a job's details
function get_job_details(uint8 jobid) public view returns(Job memory){
    return(job_details[jobid]);
}

//function to apply for a job
mapping(uint8=>uint8[]) private job_applications;
function apply_for_a_job(uint8 _JobID, uint8 _ApplicantID) public {
    job_applications[_JobID].push(_ApplicantID);

}

//mapping an applicant's rating to their ID
mapping(uint8=>uint8) public applicantrating;
// give a rating to the applicant on a scale of 1 to 5;

//function to rate an applicant (only admin can do this)

function give_rating(uint8 ID, uint8 _rating) public onlyAdmin{
    require(_rating<=5," rating can't be greater than 5");
    applicantrating[ID]=_rating;
}

//function to fetch an applicant's rating
function fetch_applicantrating(uint8 ID)public view returns(uint8){

```

```
    return(applicantrating[ID]);  
  }  
}
```