

```

class TestHireNewWorker
{
    ValidFromDateTime DefaultValidFromDateTime = HCMDateTimeUtil::startOfCurrentDay();

    FirstName firstName = 'Julia';
    LastName lastName = 'Reeves';
    HCMTitleId hcmTitleId = 'Mrs';

    HCMPositionRecId positionRecId;
    HCMWorkerRecId workerRecId;
    CompanyInfoRecId legalEntityRecId = CompanyInfo::current();
    DirPersonName personName;
    utcdatetime startDate = HCMDateTimeUtil::startOfCurrentDay();
    utcdatetime endDate = DateTimeUtil::maxValue();

    public static void main(Args _args)
    {
        TestHireNewWorker TestHireNewWorker = new TestHireNewWorker();
        TestHireNewWorker.run();
    }

    public void run()
    {
        HCMPersonnelNumberId personnelNumber = HCMWorker::getNewPersonnelNumber();

        personName = this.createPersonName(firstName, "", lastName);

        HCMHireNewWorkerContract hireNewWorkerContract =
        HCMHireNewWorkerContract::construct();
    }
}

```

```

hireNewWorkerContract.parmDirPersonName(personName);
hireNewWorkerContract.parmPersonnelNumber(personnelNumber);
// hireNewWorkerContract.parmPositionRecId(positionRecId); Assign position to worker
// hireNewWorkerContract.parmPositionAssignmentStartDate(startDate); Assign position
assignment start date
// hireNewWorkerContract.parmPositionAssignmentEndDate(endDate); Assign position
assignment end date
hireNewWorkerContract.parmEmploymentValidFrom(startDate); //Assign employment start
date
hireNewWorkerContract.parmEmploymentValidTo(endDate); //Assign employment end date
hireNewWorkerContract.parmLegalEntityRecId(legalEntityRecId); //Assign current legal entity
hireNewWorkerContract.parmEmploymentType(HcmEmploymentType::Employee);

workerRecId = HcmWorkerTransition::newHireHcmWorkerV2(hireNewWorkerContract);

HcmWorker hcmWorker = HcmWorker::find(workerRecId);

HcmTitle hcmTitle = this.createTitle(hcmTitleId);

this.createWorkerTitle(hcmWorker,hcmTitle);

}

public HcmTitle createTitle(HcmTitleId _hcmTitleId)
{
HcmTitle hcmTitle = HcmTitle::findByTitle(_hcmTitleId);

if(!hcmTitle)
{
hcmTitle.initValue();
hcmTitle.TitleId = _hcmTitleId;
hcmTitle.doInsert();
}
}

```

```
}

return hcmTitle;
}

private DirPersonName createPersonName(FirstName _firstName, MiddleName _middleName,
LastName _lastName)
{
    DirPerson dirPerson;

    dirPerson.initValue();

    DirPersonName dirPersonName;

    dirPersonName.FirstName = _firstName;
    dirPersonName.MiddleName = _middleName;
    dirPersonName.LastName = _lastName;
    dirPersonName.ValidFrom = DefaultValidFromDateTime;
    dirPersonName.ValidTo = DateTimeUtil::maxValue();

    dirPerson.updateName(dirPersonName);

    dirPerson.insert();

    dirPersonName.Person = dirPerson.ReclId;

    dirPersonName.doInsert();

    return dirPersonName;
}
```

```
public HCMWorkerTitle createWorkerTitle(HCMWorker _worker, HCMTitle _title)
{
    HCMWorkerTitle hcmWorkerTitle;

    hcmWorkerTitle.initValue();
    hcmWorkerTitle.Title = _title.ReclD;
    hcmWorkerTitle.Worker = _worker.ReclD;
    hcmWorkerTitle.ValidFrom = DateTimeUtil::minValue();
    hcmWorkerTitle.ValidTo = DateTimeUtil::maxValue();
    hcmWorkerTitle.insert();

    return hcmWorkerTitle;
}
}
```

ParagChapre.com