**Chain of Responsibility – Behavioral - 2022**

GOF: **Avoid coupling the sender of a request to its receiver by giving more than one object a chance to handle the request. Chain the receiving objects and pass the request along the chain until an object handles it.**

Diagram

Description automatically generated

Object Structure Diagam

Text

Description automatically generated with medium confidence

**Java Code Example :** A loan has be to approved based upon the amount by various actors like LoanOfficer, BranchManager and Vice President of the Bank.

**LoanData.java**

**public** **class** LoanData {

**private** String purpose;

**private** **int** amount;

**public** LoanData(String purpose, **int** amount) {

**this**.purpose = purpose;

**this**.amount = amount;

}

get()/set() methods

}

**LoanApprover.java**

**public** **abstract** **class** LoanApprover {

**protected** LoanApprover successor;

**public** **void** setSuccessor(LoanApprover successor) {

**this**.successor = successor;

}

**public** **abstract** **void** approveLoan(LoanData data);

}

**LoanOfficer.java**

**public** **class** LoanOfficer **extends** LoanApprover {

@Override

**public** **void** approveLoan(LoanData data) {

**if**(data.getAmount() < 1000 )

System.***out***.println("Loan approved by LoanOfficer ...");

**else**

successor.approveLoan(data);

}

}

**BranchManager.java**

**public** **class** BranchManager **extends** LoanApprover {

@Override

**public** **void** approveLoan(LoanData data) {

**if**( data.getAmount() > 1000 && data.getAmount() < 10000 )

System.***out***.println("Loan approved by BranchManager ...");

**else**

successor.approveLoan(data);

}

}

**VicePresident.java**

**public** **class** VicePresident **extends** LoanApprover {

@Override

**public** **void** approveLoan(LoanData data) {

System.***out***.println("Loan approved by VicePresident ...");

}

}

**Test.java**

**public** **class** Test {

**public** **static** **void** main(String[] args) {

LoanData data = **new** LoanData("Cultivation", 50000);

LoanApprover loanOfficer = **new** LoanOfficer();

LoanApprover manager = **new** BranchManager();

LoanApprover vp = **new** VicePresident();

//Form the chain

loanOfficer.setSuccessor(manager);

manager.setSuccessor(vp);

loanOfficer.approveLoan(data);

}

}