



# Certified Hyperledger Developer

Transaction Processor

# Procedure



- In the transaction processor or **logic.js** file we write the logic of the transactions defined in the model file of the project.
- First include the **decorators and metadata** into your js file which has the description of the function along with the parameters passed into the function.
- Then write the logic for the function inside the function definition mentioned in the model file.
- Each transaction type has an associated registry storing the transactions which gets updated in each function call depending on whether that transaction is allowed or not.

# EarnPoints function



```
async function EarnPoints(earnPoints) {  
  
  earnPoints.member.points = earnPoints.member.points + earnPoints.points;  
  
  const memberRegistry = await getParticipantRegistry('org.clp.biznet.Member');  
  await memberRegistry.update(earnPoints.member);  
  
  // check if partner exists on the network  
  const partnerRegistry = await getParticipantRegistry('org.clp.biznet.Partner');  
  partnerExists = await partnerRegistry.exists(earnPoints.partner.id);  
  if (partnerExists == false) {  
    throw new Error('Partner does not exist - check partner id');  
  }  
  
}
```

# UsePoints function

```
async function UsePoints(usePoints) {  
  
  if (usePoints.member.points < usePoints.points) {  
    throw new Error('Insufficient points');  
  }  
  
  usePoints.member.points = usePoints.member.points - usePoints.points  
  const memberRegistry = await getParticipantRegistry('org.clp.biznet.Member');  
  await memberRegistry.update(usePoints.member);  
  
  const partnerRegistry = await getParticipantRegistry('org.clp.biznet.Partner');  
  partnerExists = await partnerRegistry.exists(usePoints.partner.id);  
  if (partnerExists == false) {  
    throw new Error('Partner does not exist - check id');  
  }  
}
```



# THANK YOU!

Any questions?  
You can mail us at  
[hello@blockchain-council.org](mailto:hello@blockchain-council.org)