**/\*1.This function is the master function and we used it here to identify the record which was recently updated and what values were updated there.**

**2. we have called a function fieldsChanged() which field is updated in .**

**3.we have called a function fieldsChanged() which identifies what is the values updated in the record.\*/**

public static List<SObject> filterRecords(List<SObject> records, Map<Id, SObject> oldMap, List<SObjectField> fieldTokens, Map<SObjectField, String> valueMatchDef) {

List<SObject> toReturn = new List<SObject>();

for (SObject record : records) {

if (fieldsChanged(record, oldMap.get(record.Id), fieldTokens) && valuesMatch(record, valueMatchDef)) {

toReturn.add(record);

}

}

return toReturn;

}

**/\* 1.This Function used to identify the sobject's record field value changed or not.If fields values changed then it return true otherwise return false.**

**2.This fucntion is call used in the above function for checked record is updated or not \*/**

private static Boolean fieldsChanged(SObject rec, SObject old, List<SObjectField> fieldTokens) {

Boolean isChanged = false;

for (SObjectField field : fieldTokens) {

if (rec.get(field) != old.get(field)) {

isChanged = true;

break;

}

}

return isChanged;

}

/\* 1.This function is used to identify if a record is updated and then it returns true otherwise false.

2.This function is call used in the above function for checked record is updated with passed the expected value and field name in the function.\*/

\*/

private static Boolean valuesMatch(SObject record, Map<SObjectField, String> valueMatchDef) {

for (SObjectField fieldToken : valueMatchDef.keySet()) {

if (String.valueOf(record.get(fieldToken)) != valueMatchDef.get(fieldToken)) return false;

}

return true;

}