



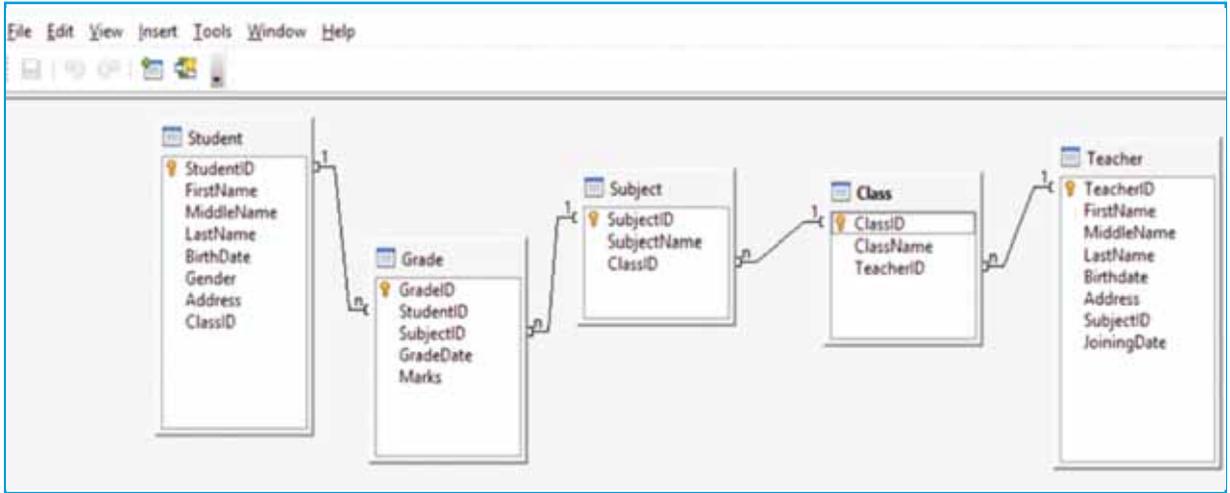
## કવેરીનો ઉપયોગ કરીને ડેટાની પુનઃપ્રાપ્તિ

### પરિચય

અગાઉના પ્રકરણોમાં, આપણે ડેટાબેઝ વિશે અને લિબ્રેઓફિસ બેઝનો ઉપયોગ કરીને ડેટાબેઝ કેવી રીતે બનાવી શકાય તે વિશે અભ્યાસ કર્યો. બેઝ ફી અને ઓપન-સોર્સ રિલેશનલ ડેટાબેઝ મેનેજમેન્ટ સિસ્ટમ છે જે ડેટાબેઝ બનાવવા માટે ગ્રાફિકલ ઇન્ટરફેસ પૂરું પાડે છે. આપણે જાણીએ છીએ કે, બેઝમાં ડેટા ટેબલ સ્વરૂપમાં સ્ટોર થાય છે. બેઝ ટેબલમાં ડેટા સ્ટોર કરવા અને ટેબલમાંથી ડેટાને પુનઃપ્રાપ્ત (રીટ્રાઈવ) કરવા માટે યુઝર ફ્રેન્ડલી પદ્ધતિ પ્રદાન કરે છે. કવેરી (Query) ભાષાનો ઉપયોગ કરીને ડેટાબેઝ સાથે સંવાદ (ઇન્ટરેક્શન) થાય છે. આ પ્રકરણમાં, આપણે ડેટાબેઝમાંથી માહિતી કેવી રીતે પુનઃપ્રાપ્ત કરવી તે વિશે શીખીશું. આપણે સૌ પ્રથમ કવેરી વિશે શીખીશું અને પછી આપણે કવેરીનો ઉપયોગ કરીને ડેટા કેવી રીતે પુનઃપ્રાપ્ત કરી શકીએ તે વિશે ચર્ચા કરીશું.

### કવેરી પ્રક્રિયા માટેનું ઉદાહરણ ડેટાબેઝ (Example database for query processing)

અગાઉના પ્રકરણમાં ચર્ચા કરેલા પગલાંને અનુસરીને, ચાલો પહેલા આકૃતિ 3.1માં બતાવ્યા મુજબ એક ડેટાબેઝ બનાવીએ. આ એક સ્ટુડન્ટ ડેટાબેઝ છે, જેમાં વિદ્યાર્થીઓ, તેમના ગ્રેડ, વિષયો, ક્લાસ અને શિક્ષકો સંબંધિત માહિતીનો સંગ્રહ કરે છે. હવે, જો આપણે જાણવા માંગીએ કે, ‘કેટલા વિદ્યાર્થીઓ 70% થી વધુ માર્ક્સ સાથે ઉત્તીર્ણ થયા છે?’ અથવા ‘કેટલા વિદ્યાર્થીઓ વર્ગ-IXમાં અભ્યાસ કરી રહ્યા છે?’, તો આપણે ડેટાબેઝમાં કવેરી કરી અને આપણા પ્રશ્નોના જવાબો મેળવી શકીએ છીએ. આગળના વિભાગમાં, આપણે ડેટાબેઝમાંથી માહિતી પુનઃપ્રાપ્ત કરવા માટે કવેરી કેવી રીતે લખવી તે વિશે શીખીશું.



આકૃતિ 3.1 : સ્ટુડન્ટ ડેટાબેઝ

### કવેરી (Query) શું છે?

ડેટાબેઝ કવેરીએ ડેટાબેઝના ડેટા પુનઃપ્રાપ્ત કરવા, સુધારવા અથવા મેનેજ કરવા માટે કરવામાં આવેલી વિનંતી છે. તે ડેટાબેઝ સાથે ક્રિયા પ્રતિક્રિયા કરવા માટેના પ્રાથમિક માધ્યમ તરીકે કાર્ય કરે છે, જે યુઝર અને એપ્લીકેશનને સંગ્રહિત માહિતીને પ્રાપ્ત કરવા અને તેમાં ફેરફાર કરવા કાર્યક્ષમ રીતે સક્ષમ બનાવે છે. મૂળભૂત રીતે, એક ડેટાબેઝ કવેરી એ કવેરી લેંગ્વેજ (સામાન્ય રીતે SQL - Structured Query Language)માં લખાયેલો એક કમાન્ડ છે, જે નીચે મુજબની ક્રિયાઓ કરવા માટે વપરાય છે, જેમ કે:

- ડેટા પુનઃપ્રાપ્ત કરવા : એક અથવા વધુ ટેબલમાંથી ચોક્કસ માહિતી પુનઃપ્રાપ્ત કરવા
- ડેટા ઇન્સર્ટ કરવા : ટેબલમાં નવા રેકોર્ડ ઉમેરવા.

- ડેટા અપડેટ કરવા : હાલના રેકોર્ડને સુધારવા.
- ડેટાને ડીલીટ કરવા : ટેબલમાંથી રેકોર્ડ ડીલીટ કરવા.

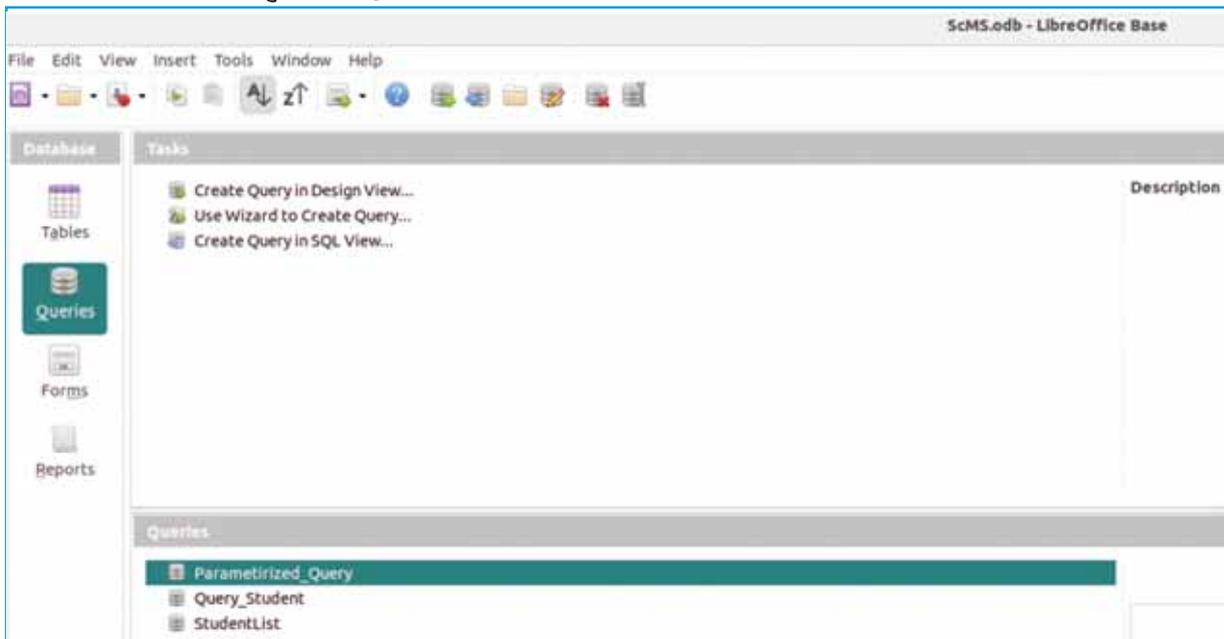
ટેબલ 3.1 મૂળભૂત SQL ક્વેરીનો સારાંશ આપે છે.

|           |  |
|-----------|--|
| રીટ્રાઈવલ | <i>SELECT * FROM STUDENT;</i><br>STUDENT ટેબલમાંથી તમામ ડેટા રીટ્રાઈવ કરે છે અને ડિસ્પ્લે કરે છે.  |
| ઇન્સર્ટ   | <i>INSERT INTO SUBJECT (SubjectID, SubjectName, ClassID)<br/>VALUES (1, Science, 9);</i><br>Subject ટેબલમાં SubjectID = 1, SubjectName = Science અને ClassID = 9 સાથે એક નવો રેકોર્ડ ઇન્સર્ટ કરે છે.   |
| અપડેટ     | <i>UPDATE SUBJECT SET ClassID = 8 WHERE SubjectID = 1;</i><br>Subject ટેબલમાં SubjectID = 1 ધરાવતી રોમાં ClassIDની વેલ્યુ બદલીને 8 કરશે.   |
| ડીલીટ     | <i>DELETE FROM SUBJECT</i><br>Subject ટેબલની તમામ રો ડીલીટ કરશે.<br><i>DELETE FROM SUBJECT WHERE SubjectName = 'Science';</i><br>SubjectName માં 'Science' છે તે તમામ રો Subject ટેબલમાંથી ડીલીટ કરશે. |

ટેબલ 3.1 : SQL ક્વેરીના ઉદાહરણો

બેઝ ક્વેરી લખવા માટે યુઝર ફ્રેન્ડલી ઇન્ટરફેસ પૂરો પાડે છે. આ ક્વેરી ઇન્ટરફેસનો ઉપયોગ કરીને, આપણે ડેટાબેઝ ટેબલમાંથી માહિતી મેળવવા માટે નિયમોનો એક સમૂહ નિર્ધારિત કરી શકીએ છીએ. બીજા શબ્દોમાં કહીએ તો, કોઈ વ્યક્તિ બેઝને જણાવી શકે છે કે, ડેટાબેઝમાંથી તે કયા ફીલ્ડ અને રેકોર્ડ જોવા માંગે છે. ક્વેરીનું પરિણામ એક ટેબલના સ્વરૂપમાં મળે છે. તે રો અને કોલમમાં ગોઠવાયેલા રેકોર્ડના સમૂહનું બનેલું હોય છે.

જ્યારે આપણે બેઝમાં કોઈ ડેટાબેઝ ખોલીએ છીએ, ત્યારે ડાબી બાજુની પેનલમાં *Queries* આઈકન દેખાય છે. ડાબી બાજુના પેનલમાં રહેલ *Queries* આઈકન પર ક્લિક કરવાથી, *Tasks* પેનલમાં ક્વેરી બનાવવા માટેની વિવિધ રીતો દર્શાવવામાં આવે છે. આકૃતિ 3.2 *Queries* આઈકન સિલેક્ટ કરવામાં આવે ત્યારે દેખાતી વિન્ડો દર્શાવે છે.



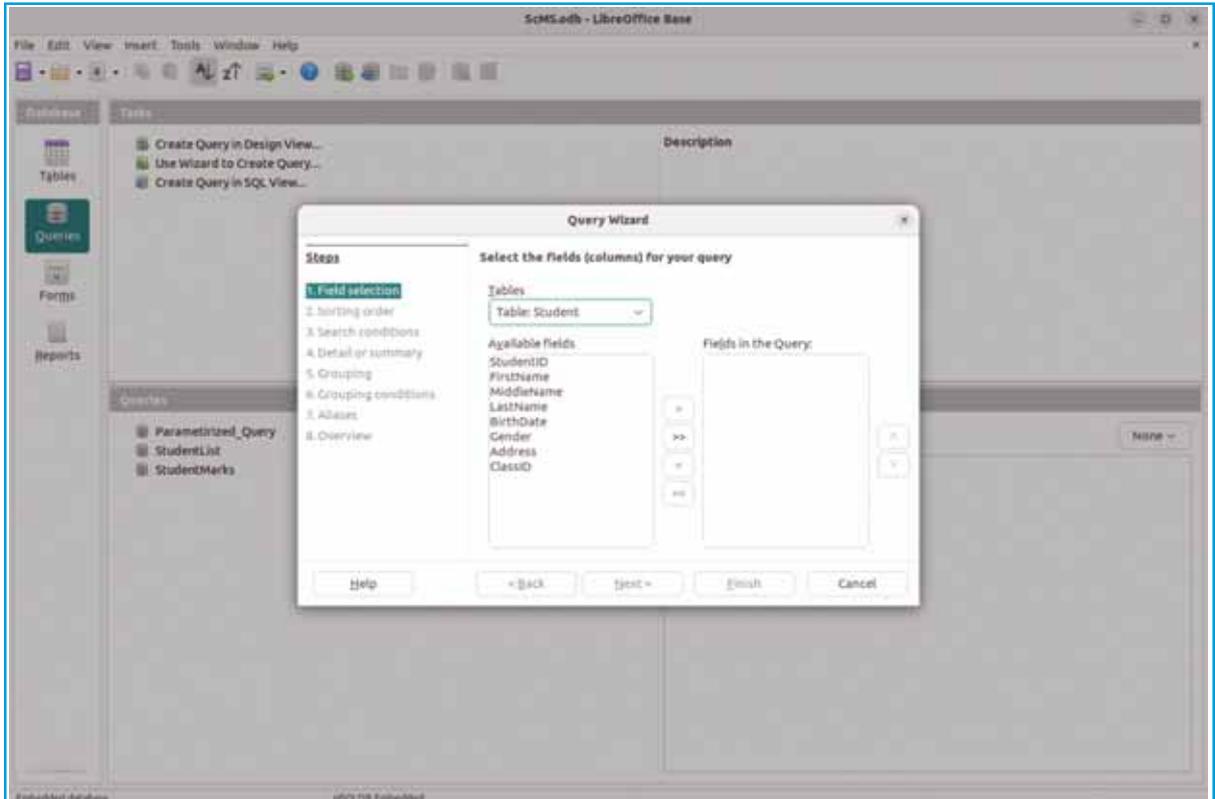
આકૃતિ 3.2 : ક્વેરી વિન્ડો

## વિઝાર્ડનો ઉપયોગ કરીને ક્વેરી બનાવવી (Creating Query Using Wizard)

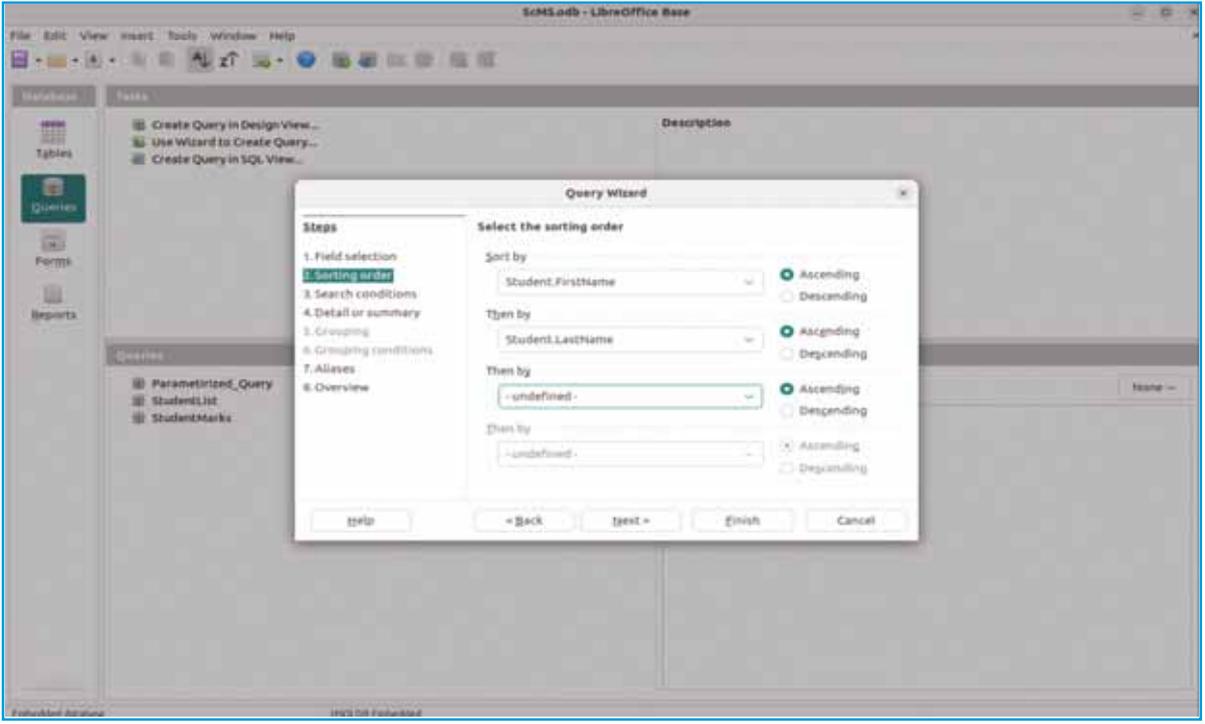
ડેટાબેઝમાં ક્વેરી બનાવવા માટે સૌથી સરળ રીત ક્વેરી વિઝાર્ડ છે. *Use Wizard to Create Query* વિકલ્પ પર ડબલ ક્લિક કરવાથી આકૃતિ 3.3 માં બતાવ્યા મુજબ ક્વેરી વિઝાર્ડ ડાયલોગ બોક્ષ ખુલે છે. ડાયલોગ બોક્ષના ડાબી બાજુની પેનલમાં જોઈ શકાય છે કે, ક્વેરી બનાવવા માટે કુલ આઠ સ્ટેપ્સની જરૂર પડે છે. જો કે, તે બધા ફરજિયાત નથી. ખરેખર તો, ફક્ત પહેલું સ્ટેપ, *Field selection* જ ફરજિયાત છે. તે આપણને આઉટપુટમાં જે ફીલ્ડ ડિસ્પ્લે થવાના છે તે ફીલ્ડને ઓળખવામાં મદદ કરે છે. બાકીના બીજા સ્ટેપ આઉટપુટને ફોર્મેટ કરવાની સવલત પૂરી પાડે છે અને જો તેની જરૂર ન હોય તો તેને છોડી શકાય છે.

**પગલું-1 :** ક્વેરી બનાવવાનું પ્રથમ સ્ટેપ એ ટેબલ અને તે ટેબલમાંથી પુનઃપ્રાપ્ત કરવાની માહિતીના ફીલ્ડના સમૂહને પસંદ કરવાનું છે. એકવાર આપણે *Tables* લેબલ હેઠળના ડ્રોપડાઉનમાંથી ટેબલ પસંદ કરીએ, પછી ડાબી બાજુનું *Available fields* લીસ્ટ બોક્ષ તે ટેબલના ફીલ્ડ દર્શાવે છે, જેનો ઉપયોગ ક્વેરીમાં થઈ શકે છે. આપણે ડાબા અને જમણા એરોનો ઉપયોગ કરીને *Available fields* માંથી *Fields in the Query* તરફ અને તેનાથી વિપરીત ફીલ્ડને મુવ કરી શકીએ છીએ. જે ફીલ્ડને આપણે આપણી ક્વેરીનો ભાગ બનાવવા માટે પસંદ કરીએ છીએ તે ફીલ્ડ *Field in the Query* લિસ્ટ બોક્ષ હેઠળ સૂચિબદ્ધ થશે. ત્યારબાદ આ ફીલ્ડને અપ અને ડાઉન બટનનો ઉપયોગ કરીને જરૂરી ક્રમમાં ગોઠવી શકાય છે. એકવાર ફીલ્ડ નક્કી થઈ જાય, પછી *Next* બટન પર ક્લિક કરો. આકૃતિ 3.3માં અવલોકન કરો કે, આપણે *Table: Student* પસંદ કર્યું છે અને તેની સાથે સંબંધિત તમામ ફીલ્ડને *Available fields* લિસ્ટ બોક્ષ હેઠળ જોઈ શકાય છે.

**પગલું-2 :** બીજું સ્ટેપ ક્વેરીના આઉટપુટને જે ક્રમમાં ડિસ્પ્લે કરવાનું છે તે માટેનું સોર્ટ ઓર્ડરનું પગલું છે. તે આપણને આઉટપુટનો સોર્ટિંગ ઓર્ડર નક્કી કરવા માટે વધુમાં વધુ ચાર ફીલ્ડ પસંદ કરવાની મંજૂરી આપે છે. ઉદાહરણ તરીકે, આપણે ક્વેરીનું પરિણામ શરૂઆતમાં *FirstName*ના ક્રમમાં અને ત્યારબાદ વિદ્યાર્થીના *LastName*ના ક્રમમાં ગોઠવેલું જોવા માંગી શકીએ છીએ. એકવાર સોર્ટિંગ ઓર્ડર નક્કી થઈ જાય, પછી *Next* બટન પર ક્લિક કરો. આકૃતિ 3.4 ફીલ્ડનો સોર્ટિંગ ઓર્ડર કેવી રીતે પસંદ કરવો તે દર્શાવે છે.

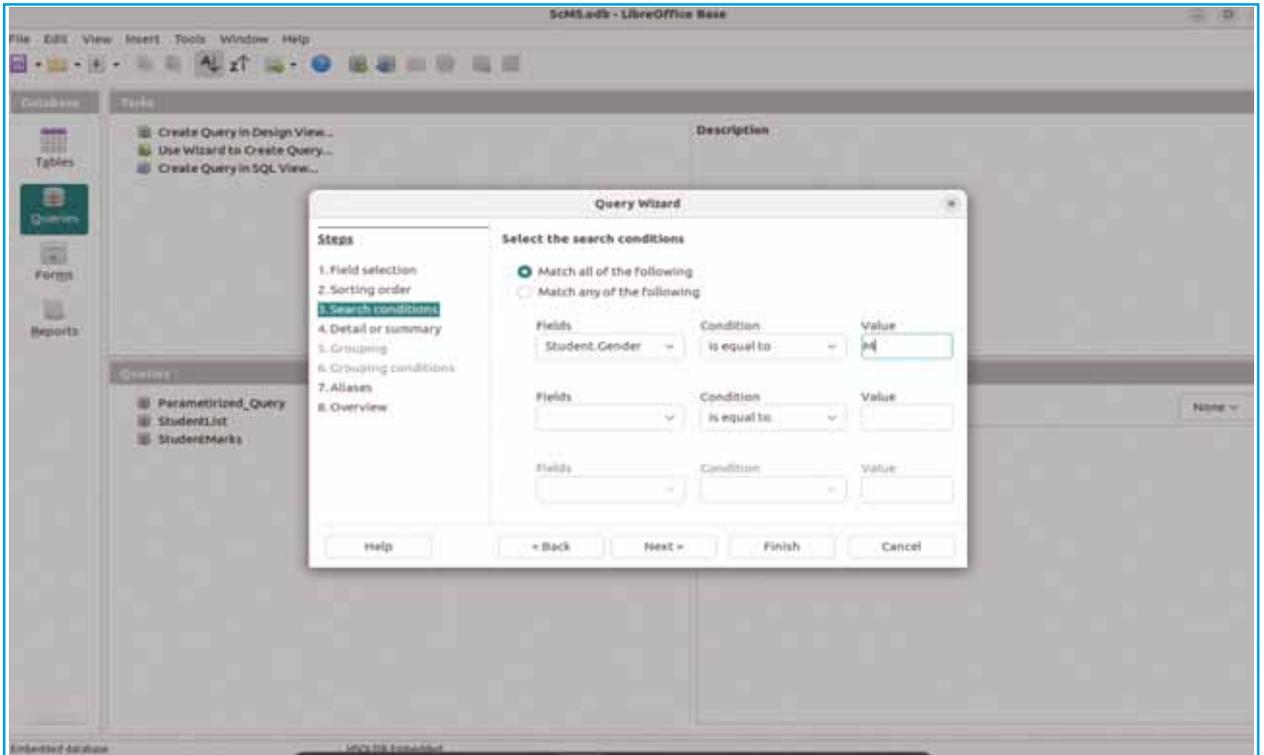


આકૃતિ 3.3 : ક્વેરી માટે ટેબલ અને ફીલ્ડની પસંદગી



આકૃતિ 3.4 : ફીલ્ડ પર સોર્ટિંગ એપ્લાઈ કરવું

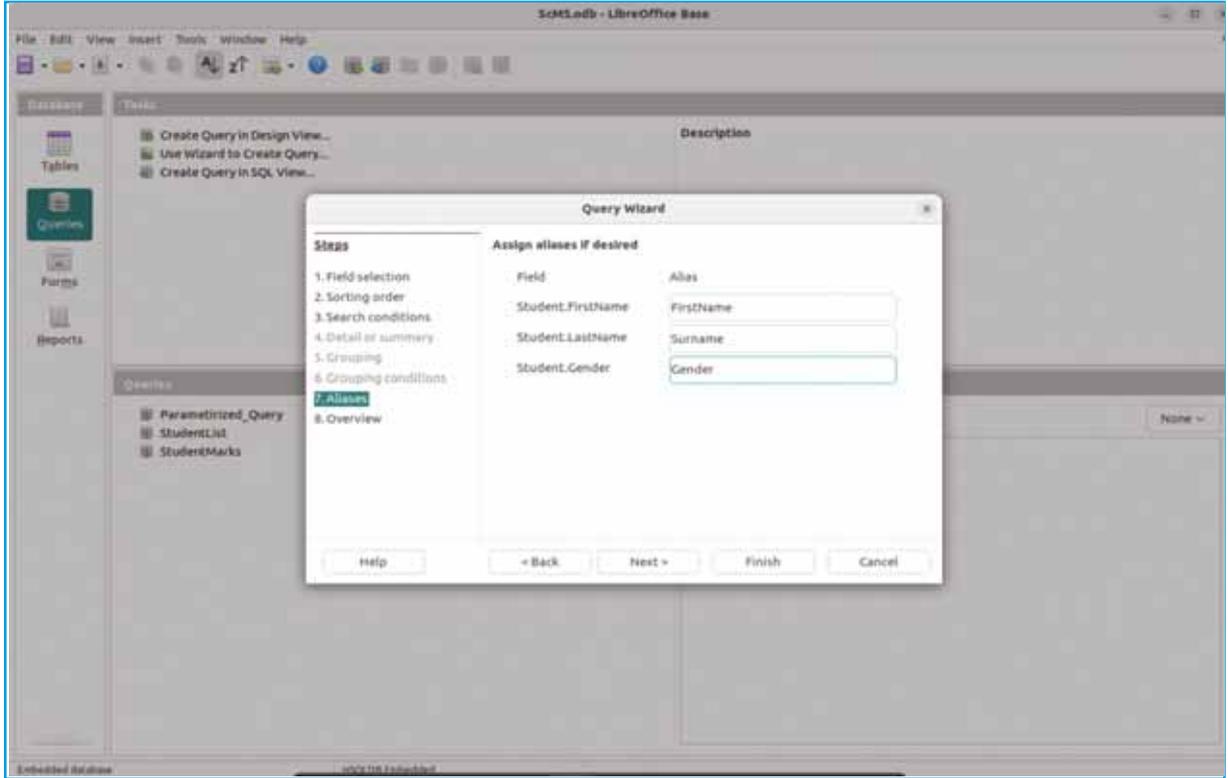
**પગલું-3 :** વિઝાર્ડના ત્રીજા સ્ટેપમાં, આપણે ક્વેરી સેટ કરીએ છીએ. અહીં આપણે *Fields*, *Condition* અને *Value* પેરામીટર્સ માટે યોગ્ય વેલ્યુ પસંદ કરવાની હોય છે. આપણે એક ક્વેરીમાં વધુમાં વધુ ત્રણ સર્ચ કન્ડિશન નિર્ધારિત કરી શકીએ છીએ. વિદ્યાર્થીઓના નામ ડિસ્કલે કરવાના કિસ્સામાં, જો આપણે માત્ર પુરુષ વિદ્યાર્થીઓનું લિસ્ટ જોઈતું હોય, તો કાર્ટેરિયા એકદમ સરળ રહેશે, આપણે *Fields* લેબલ હેઠળના ડ્રોપડાઉન માંથી *Gender* ફીલ્ડ પસંદ કરીશું. ત્યારબાદ *Condition* લેબલ હેઠળના ડ્રોપડાઉનમાંથી *is equal to* પસંદ કરીશું અને અંતે,



આકૃતિ 3.5 : ફીલ્ડ પર સર્ચ કન્ડિશન લાગુ કરવી

*Value* લેબલ હેઠળના ટેક્સ્ટ બોક્ષમાં *M* ટાઈપ કરીશું. ધ્યાન રાખો કે આ સ્ટેપમાં આપણી પાસે બે વિકલ્પો છે : *Match all of the following* અને *Match any of the following*. આપણી પાસે માત્ર એક જ કન્ડિશન હોવાથી, આપણે ડિફોલ્ટ સેટિંગ બદલવાની જરૂર નથી. જો એક કરતા વધુ સર્ચ કન્ડિશન સેટ કરવાની હોય, જેમ કે આપણે *M* અથવા *F* જેવી લિંગ ધરાવતા વિદ્યાર્થીઓને શોધી રહ્યા હોઈએ, તો આપણે *Match any of the following* પસંદ કરવાની જરૂર રહે છે. એકવાર સર્ચ કન્ડિશન નક્કી થઈ જાય પછી *Next* બટન પર ક્લિક કરો. આકૃતિ 3.5 તમામ પુરુષ વિદ્યાર્થીઓની યાદી માટેની સર્ચ કંડીશન દર્શાવે છે.

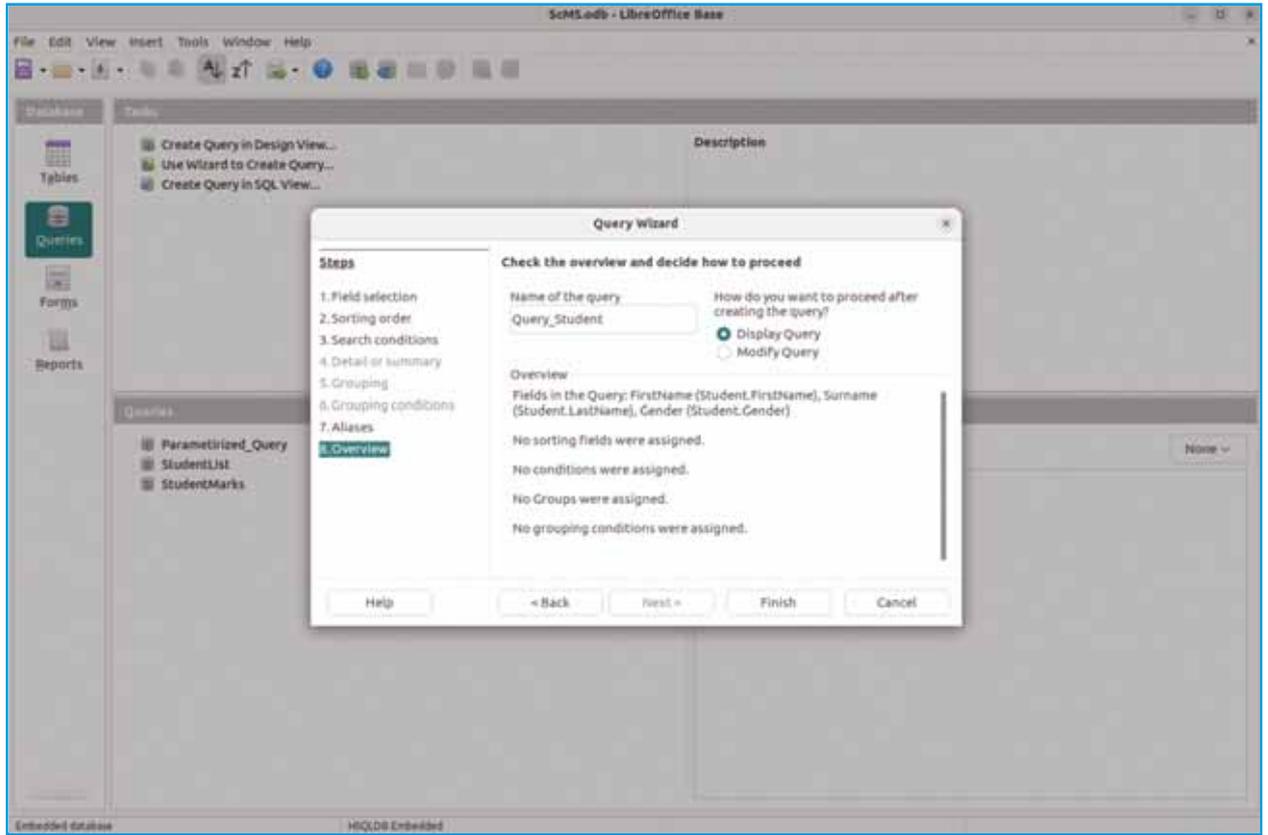
**પગલું-4, 5, 6 :** આ સ્ટેપ માહિતીને સંક્ષિપ્ત કરવા અને તેના ગ્રુપ કરવા માટેના વિકલ્પોને લગતા છે. આપણી ઉદાહરણ ક્વેરીમાં, પસંદ કરેલ *Student* ટેબલમાં કોઈ ન્યુમેરિક ફીલ્ડ નથી અને તેથી માહિતીને સંક્ષિપ્ત કરવા અથવા આંકડાકીય ગણતરીઓ કરવાના વિકલ્પો ધરાવતા સ્ટેપને છોડી દેવામાં આવ્યા છે.



### આકૃતિ 3.6 : ઉપનામ ઉમેરવા

**પગલું - 7 :** સાતમા સ્ટેપમાં, બેઝ પસંદ કરેલા ફીલ્ડના નામો માટે ઉપનામો આપવાની સગવડ આપે છે. સામાન્ય રીતે ડેટાબેઝ પ્રોગ્રામર દ્વારા આપવામાં આવેલ વાસ્તવિક ફીલ્ડ નામ યુઝર માટે સરળતાથી સમજી શકાય તેવા ફોર્મેટમાં ન પણ હોય. ઉદાહરણ તરીકે, ડેટાબેઝ ફીલ્ડ નેમમાં *first\_name* જેમાં કેટલાક સ્પેશીયલ કેરેક્ટર હોઈ શકે છે, અથવા તે *fName*; જેવું સંક્ષિપ્ત હોઈ શકે છે. આવા કિસ્સાઓ માટે, આપણે *FirstName* જેવું સહજ રીતે સમજી શકાય તેવું ઉપનામ આપી શકીએ છીએ. આમ, ઉપનામ બનાવવાનો હેતુ એ છે કે ક્વેરી વિઝાર્ડ ફીલ્ડના નામને વાંચવામાં સરળ રહે તેવા સ્વરૂપમાં ડિસ્પ્લે કરે છે. આ સ્ટેપ પણ વૈકલ્પિક છે; જો જરૂરી હોય તો જ આપણે ઉપનામ ઉમેરી શકીએ છીએ. જો આપણે ઉપનામ ઉમેરીશું નહીં તો, ટેબલના ફીલ્ડના નામ તે જ રીતે ડિસ્પ્લે થશે. એકવાર બધા ઉપનામ નક્કી થઈ જાય, પછી *Next* બટન પર ક્લિક કરો. આકૃતિ 3.6 ઉપનામ કેવી રીતે ઉમેરવા તે દર્શાવે છે.

**પગલું - 8 :** અંતે, આઠમા સ્ટેપમાં, આપણે અત્યાર સુધી કરેલા તમામ સ્ટેપની એક ઝાંખી દર્શાવવામાં આવે છે. આકૃતિ 3.7 હાલમાં બનાવેલ ક્વેરીની ઝાંખી દર્શાવે છે. *Name of the query* લેબલવાળા ટેક્સ્ટ બોક્ષમાં ક્વેરીનું નામ ટાઈપ કરો. આપણા કિસ્સામાં, આપણે તેને *Query\_Student* નામ આપ્યું છે. આપણે શું કર્યું છે તેના પર એક ક્ષણ માટે ધ્યાન આપો. જો કોઈ ફેરફાર કરવાના હોય, તો *Back* બટનનો ઉપયોગ કરીને જરૂરી ફેરફારો કરો.



આકૃતિ 3.7 : બનાવેલ ક્વેરીની ઝાંખી

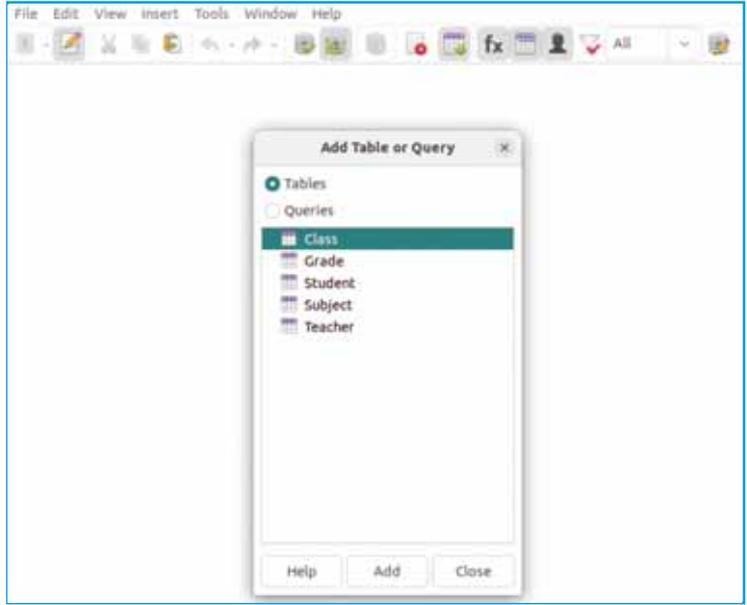
આપણે *Finish* બટન પર ક્લિક કરીશું ત્યારે ક્વેરીનું પરિણામ આકૃતિ 3.8માં બતાવ્યા મુજબ ડેટાશીટ વ્યૂમાં ડિસ્પ્લે થશે. (નોંધ લો કે ‘એન્થોની ગોમ્સ, રફીક મેનન અને સની જૈન’ નામો ડેટાબેઝ ટેબલમાંથી મેળવેલ વેલ્યુ છે. જો આપણે “રમેશ, સુરેશ” જેવા અન્ય નામો ઇન્સર્ટ કર્યા હશે, તો તે નામો બતાવશે.)

|   | FirstName | LastName | Gender |
|---|-----------|----------|--------|
| ▶ | Anthony   | Gomes    | M      |
|   | Rafiq     | Memon    | M      |
|   | Sunny     | Jain     | M      |

આકૃતિ 3.8 : ક્વેરીનું પરિણામ

## ડિઝાઈન વ્યૂનો ઉપયોગ કરીને ક્વેરી બનાવવી

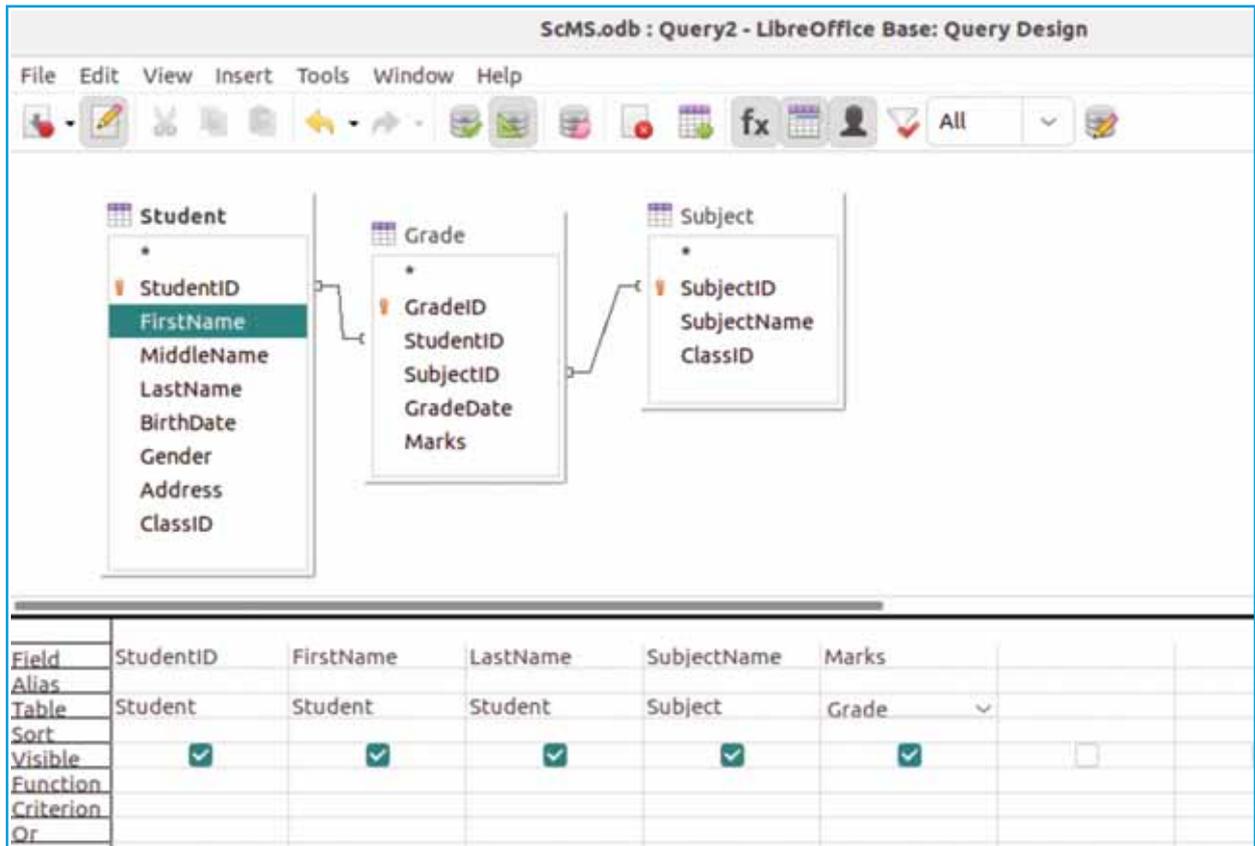
*Query Wizard* કે જેનો ઉપયોગ આપણે અગાઉના વિભાગમાં કર્યો છે, તે સરળ છે અને નવા શીખનારાઓ માટે ઉપયોગી છે. બેઝમાં ક્વેરી બનાવવા માટે ડિઝાઈન વ્યૂ બીજો એક વિકલ્પ છે. જ્યારે *Query Wizard* એ મૂળભૂત ક્વેરી બનાવવા માટે એક માર્ગદર્શક, સ્ટેપ-બાય-સ્ટેપ ટુલ છે અને *Query Design View* ક્વેરી બનાવવા અને તેમાં ફેરફાર કરવા માટે વધુ અદ્યતન નિયંત્રણ અને અનુકૂળન કે ફેરફારના વિકલ્પો પ્રદાન કરે છે. વિઝાર્ડ સરળ ક્વેરી માટે અને એવા યુઝર્સ માટે મદદરૂપ છે કે જેઓ વધુ માર્ગદર્શિત અભિગમ પસંદ કરે છે, જ્યારે ડિઝાઈન વ્યૂ જટિલ ક્વેરી માટે અથવા જ્યારે વધુ બારીક નિયંત્રણની જરૂર હોય ત્યારે ઉપયોગી છે.



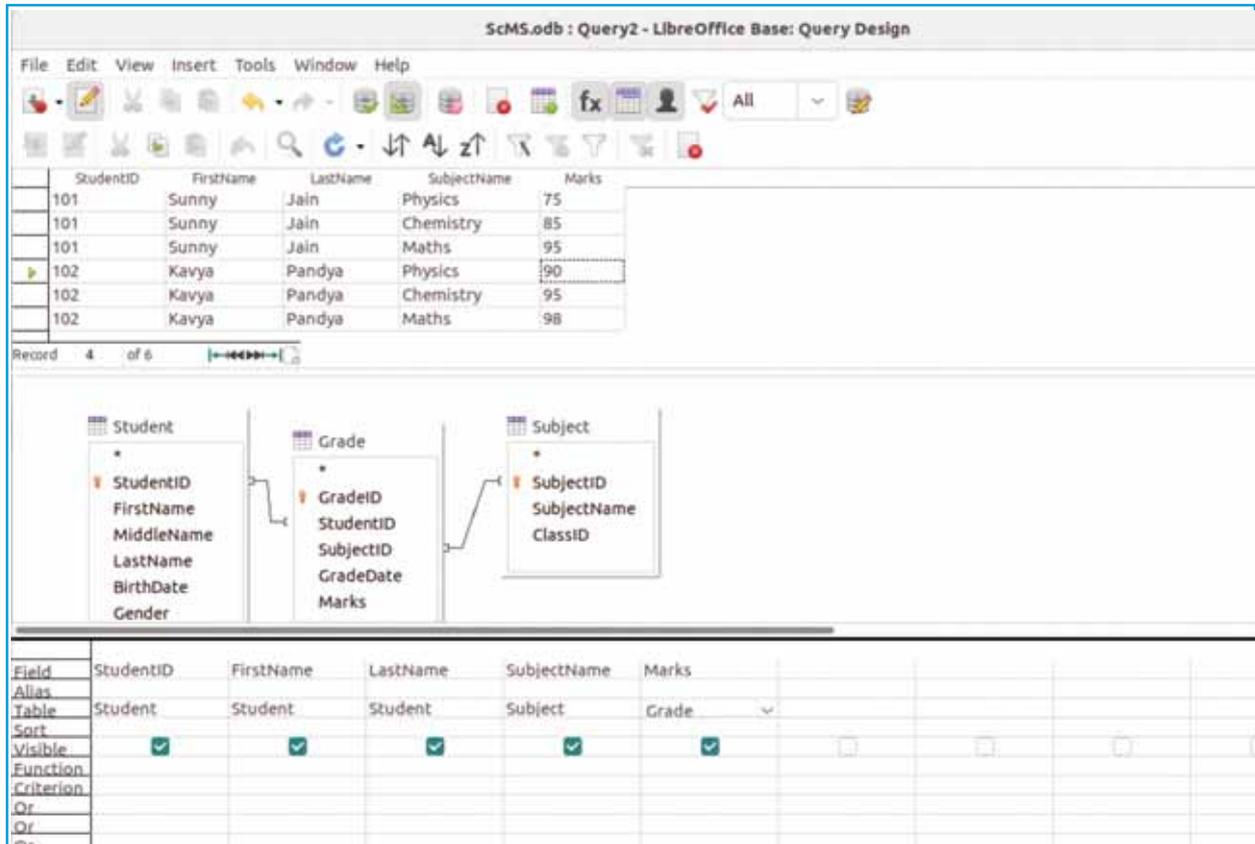
આકૃતિ 3.9 : Add ટેબલ ઓર ક્વેરી ડાયલોગ બોક્સ

હવે આપણે ડિઝાઈન વ્યૂનો ઉપયોગ કરીને એક ક્વેરી બનાવીશું. આકૃતિ 3.2માં બતાવેલ ક્વેરી ટેબમાં, *Create Query in Design View* પર ક્લિક કરો. આમ કરવાથી આકૃતિ 3.9માં બતાવ્યા મુજબ *Add Table or Query* ડાયલોગ બોક્સ ખુલશે.

- *Student* ટેબલને સિલેક્ટ કરો અને *Add* બટન પર ક્લિક કરો.
- તે જ રીતે, *Grade* અને *Subject* ટેબલને સિલેક્ટ કરો. આકૃતિ 3.10માં બતાવ્યા પ્રમાણે આપણે ટેબલ પેનમાં ત્રણ ટેબલ જોઈ શકીએ છીએ. બેઝ રીલેશનશીપ પણ ડિસ્પ્લે કરે છે.
- *Add Table or Query* ને બંધ કરવા માટે *Close* બટન પર ક્લિક કરો.
- *Student* ટેબલમાંથી *StudentID*, *FirstName*, અને *LastName* પર ડબલ ક્લિક કરો. તે જ રીતે, *Subject* ટેબલમાંથી *SubjectName* ફીલ્ડ અને *Grade* ટેબલમાંથી *Marks* ફીલ્ડ પસંદ કરો. આકૃતિ 3.10માં બતાવ્યા પ્રમાણે ફીલ્ડના નામ તેમના સંબંધિત ટેબલના નામ સાથે ગ્રીડમાં ડિસ્પ્લે થશે.
- નોંધ લો કે આપણે *Alias*, *Sort*, *Visible*, *Function*, *Criterion* અને *Or* જેવા કેટલાક રેકોર્ડ (રો) શીર્ષકો પણ જોઈ શકીએ છીએ. મૂળભૂત રીતે દરેક ફીલ્ડ માટે *Visible* ઓપ્શન ટૂંક થયેલું હોય છે. તે સૂચવે છે કે તમામ સિલેક્ટ કરેલા ફીલ્ડ આઉટપુટમાં ડિસ્પ્લે થશે.
- અગાઉના વિભાગમાં ચર્ચા કર્યા મુજબ, ફીલ્ડ માટે અર્થપૂર્ણ નામ ડિસ્પ્લે કરવા માટે ઉપનામ (*Alias*)નો ઉપયોગ કરી શકાય છે. ઉદાહરણ તરીકે, *Firstname*ને બદલે, આપણે ક્વેરીના પરિણામમાં કોલમ ટાઈટલ તરીકે *Name of Student* નો ઉપયોગ કરવાનું પસંદ કરીશું. *FirstName* કોલમ હેઠળ *Alias* રો શીર્ષક પછી દેખાતા ટેક્સ્ટ બોક્સમાં *Name of Student* ટાઈપ કરો.
- સોર્ટિંગ આપણને વેલ્યુના ચડતા અથવા ઉતરતા ક્રમમાં રેકોર્ડ ડિસ્પ્લે કરવાની સવલત આપે છે. વિદ્યાર્થીઓના રેકોર્ડને નામોના મૂળાક્ષર ક્રમમાં ડિસ્પ્લે કરવા માટે, *FirstName* કોલમ હેઠળના *Sort* ઓપ્શનમાં *ascending* પસંદ કરો. તે જ રીતે, *StudentID* નંબરના ચડતા ક્રમમાં રેકોર્ડને ડિસ્પ્લે કરવા માટે, *StudentID* કોલમ હેઠળ *Sort* રો હેડિંગ પછી દેખાતા ડ્રોપ ડાઉન બોક્સમાંથી *ascending* પસંદ કરો.
- *Query Design* ટૂલબાર પરના *Run Query* બટન પર ક્લિક કરો. આકૃતિ 3.11 માં બતાવ્યા મુજબનું ક્વેરીનું પરિણામ ડિસ્પ્લે થશે.
- ક્વેરીને સેવ કરવા માટે, *File* મેનુમાંથી *Save* ઓપ્શન પસંદ કરો. વૈકલ્પિક રીતે, *Close* બટન પર ક્લિક કરો એટલે બેઝ એક *Save* ડાયલોગ બોક્સ ડિસ્પ્લે કરશે.
- ઈચ્છિત નામ ટાઈપ કરો, ઉદાહરણ તરીકે *StudentMarks*, અને *OK* બટન પર ક્લિક કરો.



આકૃતિ 3.10 : ફીલ્ડની પસંદગી



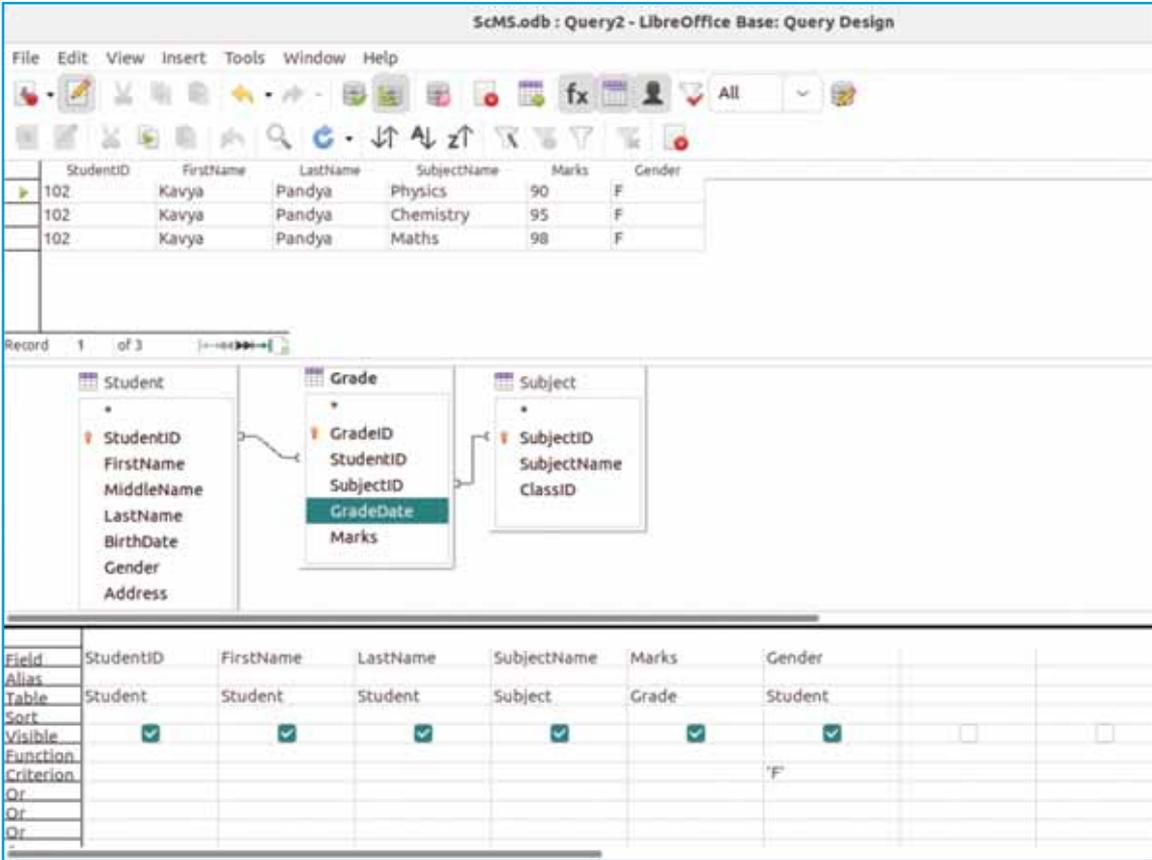
આકૃતિ 3.11 : ક્વેરીનો આઉટપુટ

## માપદંડ સાથેની ક્વેરી (Queries with Criteria)

આપણે જોયું છે કે આપણે એવી ક્વેરી લખી શકીએ છીએ, જે ટેબલના પસંદ કરેલા ફીલ્ડને ડિસ્પ્લે કરે છે. હવે, ધારો કે, તમામ રેકોર્ડ જોવાની જગ્યાએ, આપણે એવા રેકોર્ડ જોવા માંગીએ છીએ જે અમુક ચોક્કસ માપદંડને પૂર્ણ કરે છે. ઉદાહરણ તરીકે, માત્ર સ્ત્રી વિદ્યાર્થી (ફીમેલ સ્ટુડન્ટ)ની વિગતો. આનો અર્થ એ છે કે આપણે સિલેક્ટ કરેલા રેકોર્ડનું એક પેટા-જૂથ ડિસ્પ્લે કરવા માંગીએ છીએ. આ માટે, આપણે એક માપદંડ નક્કી કરી શકીએ છીએ જે રેકોર્ડને માત્ર *Gender* ફીલ્ડમાં 'F' મૂલ્ય ધરાવતા રેકોર્ડ પૂરતા મર્યાદિત કરશે.

## એક ફીલ્ડનો ઉપયોગ (Using Single Field)

*Gender* ફીલ્ડના *Criterion* સેલમાં આકૃતિ 3.12માં બતાવ્યા મુજબ 'F' ટાઈપ કરો. નોંધ લો કે, ક્રાઈટેરિયા ટેક્સ્ટને ક્વોટેશન (") ડેલીમીટરની અંદર લખવું આવશ્યક છે, જ્યારે નંબરને કોઈપણ ડેલીમીટરની જરૂર નથી. જો આપણે ડેલીમીટર ન મૂકીએ, તો બેઝ કોઈ ભૂલ દેખાડશે નહીં; તેના બદલે તે જાતે જ ડેલીમીટર લાગુ કરશે. અગાઉ ચર્ચા કર્યા મુજબ ક્વેરીને *Save* અને રન કરીએ એટલે આપણને આઉટપુટ તરીકે સ્ત્રી વિદ્યાર્થીની યાદી જોવા મળશે.



આકૃતિ 3.12 : માપદંડ સેટ કરવા

## લોજિકલ ઓપરેટર (Logical Operators) નો ઉપયોગ કરવો

આકૃતિ 3.12માં દર્શાવેલ કોન્સ્ટન્ટ વેલ્યુના ઉપયોગની ઉપરાંત, બેઝ આપણને વિવિધ લોજિકલ ઓપરેટરના આધારે ક્રાઈટેરિયા નક્કી કરવા માટેની પણ મંજૂરી આપે છે, જેમ કે, જે વિદ્યાર્થીઓના ગ્રેડ 60% થી વધુ હોય તેમના નામો પ્રિન્ટ કરવા.

ધારો કે આપણે એવા વિદ્યાર્થીઓની યાદી ડિસ્પ્લે કરવા માંગીએ છીએ કે જેમનો જન્મ 1<sup>st</sup> જાન્યુઆરી 2001 પછી થયો હોય. *Design View* માં એક નવી ક્વેરી બનાવો. *Student* ટેબલ ઉમેરો. *BirthDate* ફીલ્ડના *Criterion* સેલમાં "> #01/01/2001#" ટાઈપ કરો. હવે, ક્વેરી ને સેવ કરો અને રન કરો. આઉટપુટ આકૃતિ 3.13 જેવું જ દેખાશે.

ScMS.odb : Query2 - LibreOffice Base: Query Design

File Edit View Insert Tools Window Help

102 Kavya Pandya Physics 90 F 20/01/02  
 102 Kavya Pandya Chemistry 95 F 20/01/02  
 102 Kavya Pandya Maths 98 F 20/01/02

Record 1 of 3

**Student**  
 StudentID  
 FirstName  
 MiddleName  
 LastName  
 BirthDate  
 Gender  
 Address

**Grade**  
 GradeID  
 StudentID  
 SubjectID  
 GradeDate  
 Marks

**Subject**  
 SubjectID  
 SubjectName  
 ClassID

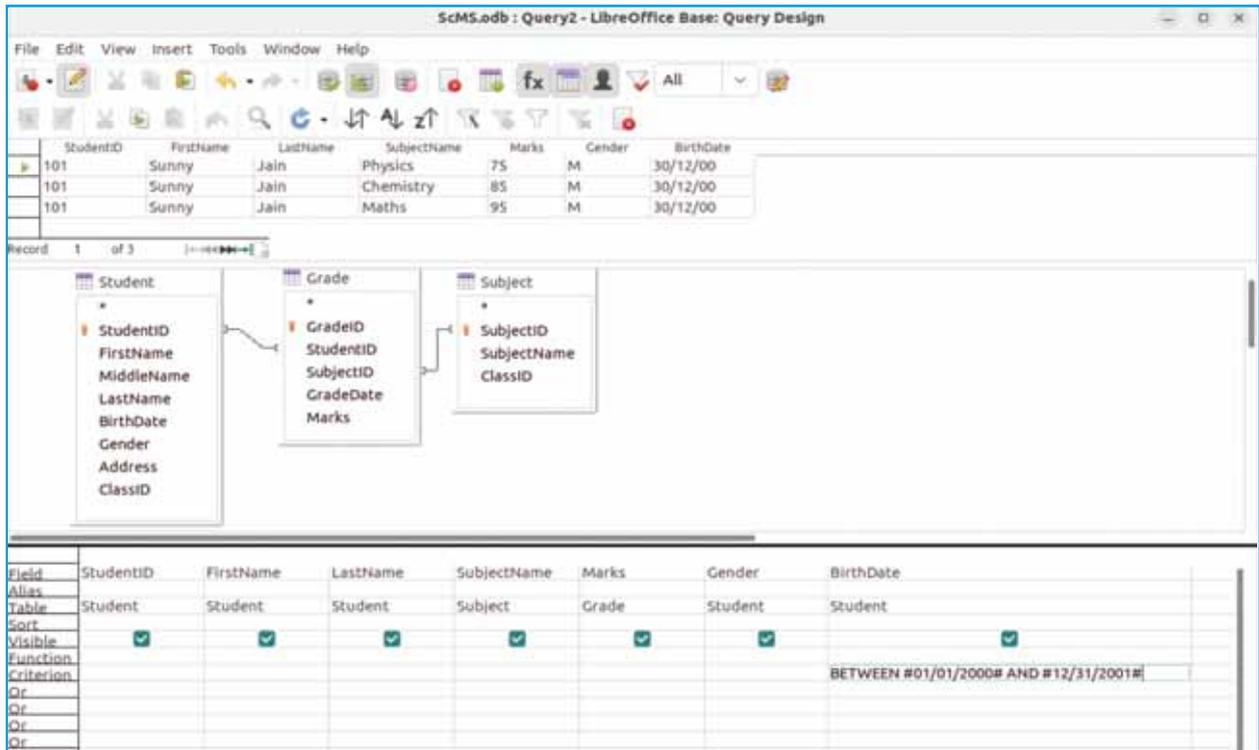
| Field     | StudentID | FirstName | LastName | SubjectName | Marks | Gender  | BirthDate       |
|-----------|-----------|-----------|----------|-------------|-------|---------|-----------------|
| Alias     |           |           |          |             |       |         |                 |
| Table     | Student   | Student   | Student  | Subject     | Grade | Student | Student         |
| Sort      |           |           |          |             |       |         |                 |
| Visible   | ✓         | ✓         | ✓        | ✓           | ✓     | ✓       | ✓               |
| Function  |           |           |          |             |       |         |                 |
| Criterion |           |           |          |             |       |         | >= #01/01/2001# |
| Or        |           |           |          |             |       |         |                 |
| Or        |           |           |          |             |       |         |                 |
| Or        |           |           |          |             |       |         |                 |

### આકૃતિ 3.13 : Date ફીલ્ડમાં કાઈટેરિયા લાગુ કરવો

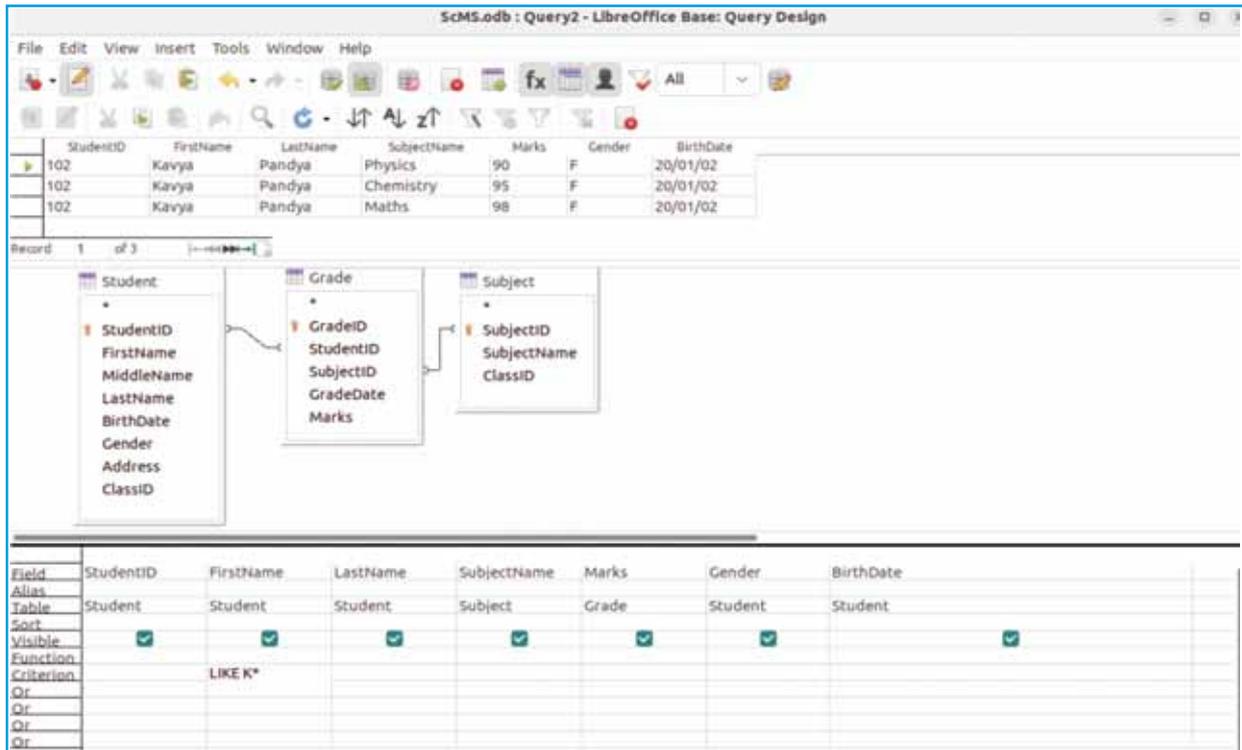
તે જ રીતે, 1st જાન્યુઆરી 2000 અને 31st ડિસેમ્બર 2001ની વચ્ચે આવતા વિદ્યાર્થીઓને ડિસ્પ્લે કરવા માટે, *BirthDate* ફીલ્ડના Criterion ને “>= # 01/01/2000 # AND <= # 12/31/2001 #”. તરીકે સેટ કરી શકાય છે. આકૃતિ 3.14માં બતાવ્યા પ્રમાણે, આ જ કાઈટેરિયા સેટ કરવા માટે બેઝ *BETWEEN* ઓપરેટર પણ આપે છે.

### વાઈલ્ડ કાર્ડ (Wild Card)નો ઉપયોગ

ધારો કે આપણે એવા વિદ્યાર્થીઓની યાદી જોવા માંગીએ છીએ જેમનું પ્રથમ નામ *K* મૂળાક્ષરથી શરૂ થતું હોય. આ કરવા માટે, *Student* ટેબલનો ઉપયોગ કરીને એક નવી ક્વેરી બનાવો. આકૃતિ 3.15માં બતાવ્યા મુજબ *Criterion* ને *LIKE 'K\*'* તરીકે સેટ કરો. આકૃતિ 3.15માં *FirstName* ફીલ્ડના *Criterion* સેલમાં એક્સપ્રેશનમાં ઉપયોગમાં લેવાયેલ એસ્ટેરિસ્ક (\*) ને વાઈલ્ડકાર્ડ કેરેક્ટર તરીકે ઓળખવામાં આવે છે. વાઈલ્ડકાર્ડ એક સિમ્બોલ છે કે જે કોઈપણ એક અક્ષર અથવા અક્ષરના સમૂહનું પ્રતિનિધિત્વ કરે છે. ‘K\*’ એવા શબ્દનું પ્રતિનિધિત્વ કરે છે જેનો પ્રથમ અક્ષર *K* છે, અને ત્યારબાદ કોઈપણ અક્ષરોનો સમૂહ હોઈ શકે છે. તે જ રીતે, *LIKE '\*k'* કાઈટેરીયા એવા વિદ્યાર્થીઓને ડિસ્પ્લે કરશે જેના નામો ‘k’ અક્ષરથી સમાપ્ત થતા હોય. આવી જ રીતે આપણે એક અક્ષર માટે ? વાઈલ્ડકાર્ડ કેરેક્ટરનો ઉપયોગ કરી શકીએ.



આકૃતિ 3.14 : Between ઓપરેટરનો ઉપયોગ



આકૃતિ 3.15 : વાઈલ્ડ કાર્ડ્સ

### પેરામીટર (Parameters) સાથેની ક્વેરી

અત્યાર સુધી, આપણે જે ક્વેરી બનાવી છે તેમાં નિશ્ચિત માપદંડનો ઉપયોગ થતો હતો. એકવાર આપેલ માપદંડ ક્વેરીના દરેક અમલ (એક્ઝિક્યુશન) વખતે બદલાશે નહીં. જોકે, ક્વેરીનું આઉટપુટ ટેબલમાં રહેલ ડેટાના આધારે બદલાઈ શકે છે.

પેરામીટર ક્વેરીને રન ટાઈમ યુઝર પાસેથી વેલ્યુ સ્વીકારવા માટે ડિઝાઈન કરવામાં આવે છે. સામાન્ય રીતે, જ્યારે આપણે પેરામીટર ક્વેરી રન કરીએ છીએ, ત્યારે તે યુઝરને પેરામીટરની વેલ્યુ એન્ટર કરવા માટે એક ડાયલોગ બોક્ષ ડિસ્પ્લે કરશે. આ વેલ્યુ પછી ડેટા પુનઃપ્રાપ્ત કરવા માટે માપદંડની (કાઈટેરીયન) વેલ્યુ તરીકે એસાઈન કરવામાં આવે છે.

ચાલો, આપણે Student ટેબલમાં ઉપલબ્ધ વિદ્યાર્થીઓની વિગતો દર્શાવવા માટે એક ક્વેરી બનાવીએ. નીચે આપેલા પગલાંનો ઉપયોગ કરવાથી આપણને ઈચ્છિત પરિણામ મળશે.

- નવી ક્વેરી ડિઝાઈન વ્યૂમાં ખોલો.
- Student ટેબલ ઉમેરો.
- Address ફીલ્ડના Criterion સેલમાં : Address અથવા [Address] ટાઈપ કરો.

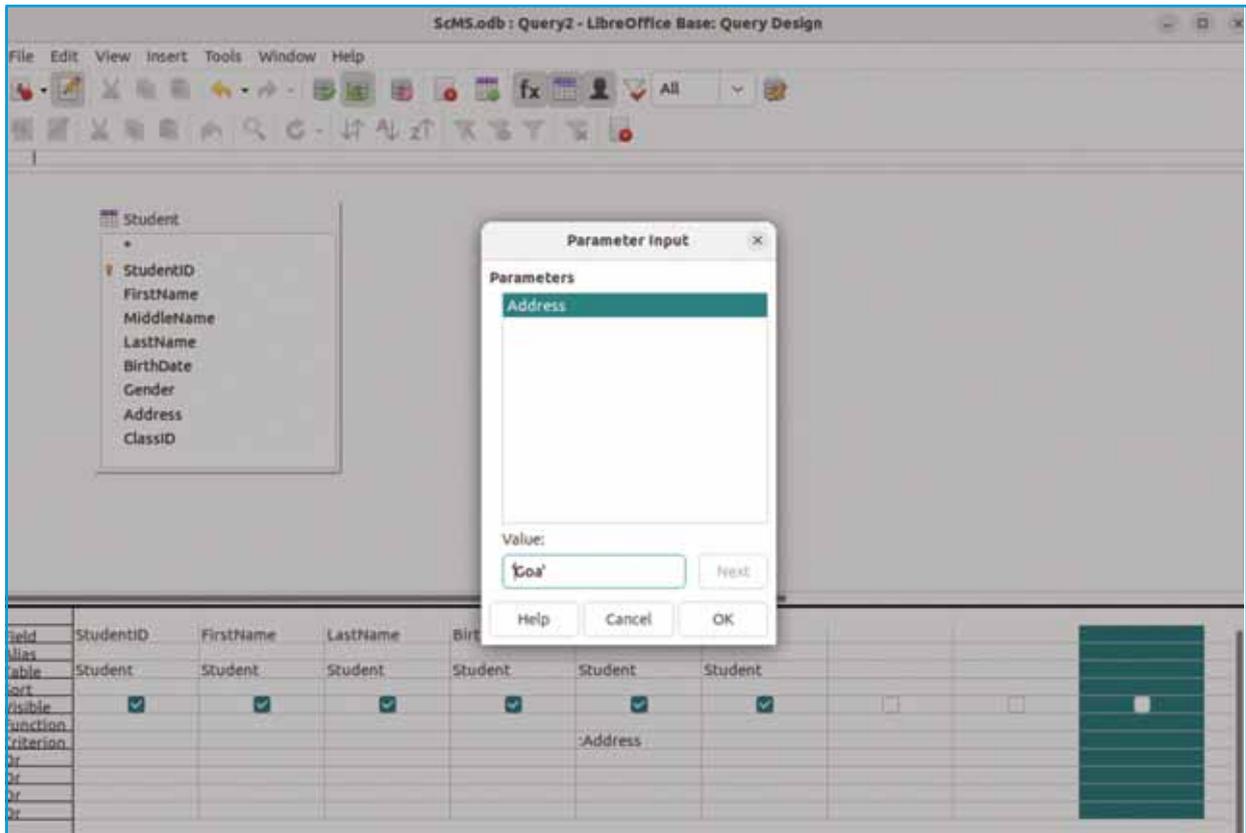
ક્વેરી આકૃતિ 3.16માં દર્શાવ્યા પ્રમાણે દેખાશે. નોંધ લો કે માપદંડ માટેના પેરામીટરની પહેલાં કોલન સિમ્બોલ (: ) હોવો જરૂરી છે.

- ક્વેરીનું પરિણામ જોવા માટે Run બટન પર ક્લિક કરો. બેઝ આકૃતિ 3.17માં દર્શાવ્યા પ્રમાણે એક ડાયલોગ બોક્ષ ડિસ્પ્લે કરશે.
- Value લેબલની નીચે આપેલા ટેક્સ્ટબોક્સમાં 'Goa' ટાઈપ કરો અને OK પર ક્લિક કરો. આકૃતિ 3.18માં દર્શાવ્યા પ્રમાણે આપણને 'Goa' માં રહેતા વિદ્યાર્થીઓની યાદી મળશે.

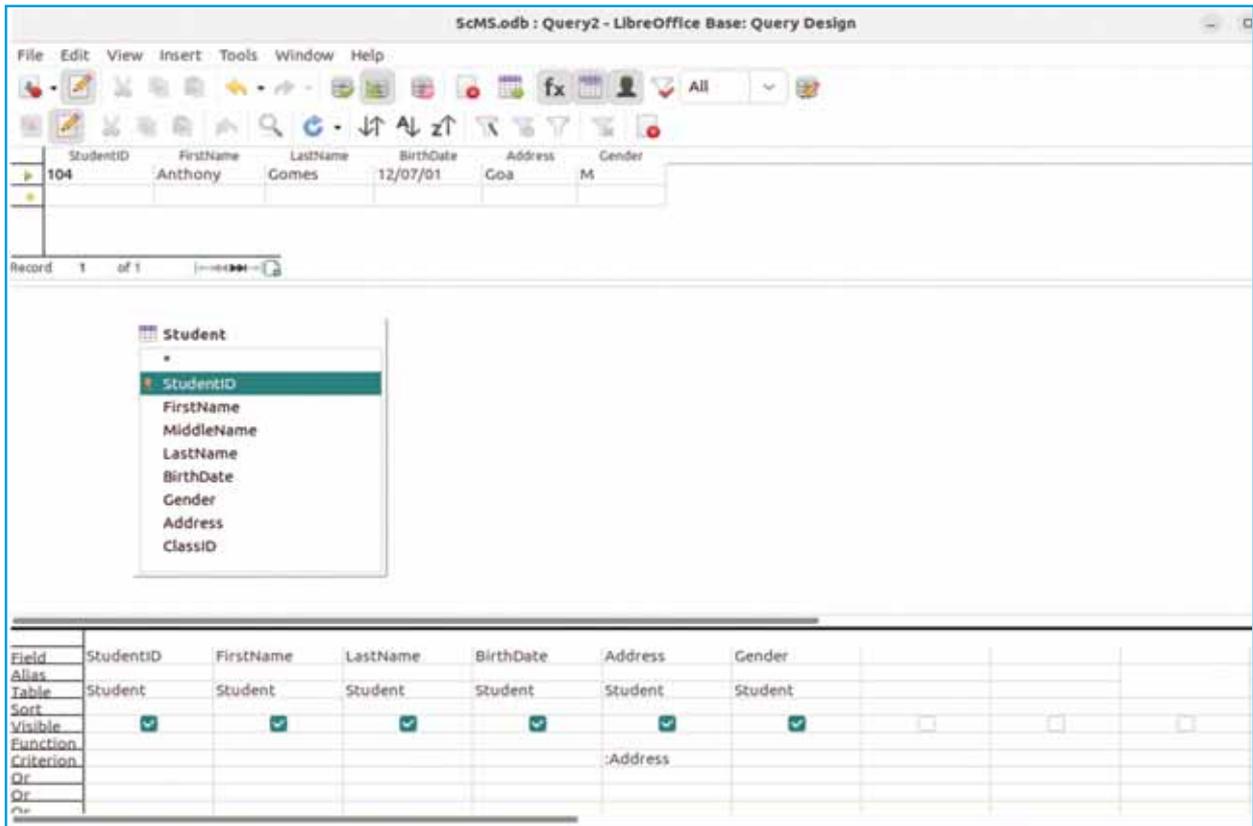
The screenshot shows the 'Query Design' view in LibreOffice Base. The 'Student' table is selected, and the 'Address' field is highlighted in the field list. Below the field list, a table shows the design grid for the query.

| Field     | StudentID | FirstName | LastName | BirthDate | Address  | Gender  |
|-----------|-----------|-----------|----------|-----------|----------|---------|
| Alias     |           |           |          |           |          |         |
| Table     | Student   | Student   | Student  | Student   | Student  | Student |
| Sort      |           |           |          |           |          |         |
| Visible   | ✓         | ✓         | ✓        | ✓         | ✓        | ✓       |
| Function  |           |           |          |           |          |         |
| Criterion |           |           |          |           | :Address |         |
| Or        |           |           |          |           |          |         |
| Or        |           |           |          |           |          |         |

આકૃતિ 3.16 : પેરામીટરવાળી ક્વેરી



आकृति 3.17 : पेरामीटर वेव्यु



आकृति 3.18 : आउटपुट

## સારાંશ

એક ડેટાબેઝ સિસ્ટમ મોટા પ્રમાણમાં ડેટા સંગ્રહિત કરવાની એક પદ્ધતિ પૂરી પાડે છે. ડેટાબેઝ સિસ્ટમમાંથી અંતિમ યુઝર માટે ઉપયોગી ડેટા પુનઃપ્રાપ્ત કરવો એક પડકારજનક કાર્ય છે. ડેટાબેઝ ક્વેરી યુઝરને ડેટાબેઝમાંથી અર્થપૂર્ણ માહિતી પુનઃપ્રાપ્ત કરવાની મંજૂરી આપે છે. SQL એ ડેટાબેઝ ટેબલમાંથી ડેટા પુનઃપ્રાપ્ત કરવાની એક ભાષા છે. SQLનો ઉપયોગ કરીને, આપણે એક કરતાં વધુ ટેબલમાંથી પણ ડેટા પુનઃપ્રાપ્ત કરી શકીએ છીએ. લિબ્રેઓફીસ બેઝ ડેટાબેઝ ક્વેરી લખવા માટે યુઝર-ફ્રેન્ડલી ઈન્ટરફેસ પ્રદાન કરે છે. ક્વેરી વિઝાર્ડ વ્યૂ ડેટાબેઝ ક્વેરી બનાવવા માટે સ્ટેપ-બાય-સ્ટેપ માર્ગદર્શિત અભિગમ પૂરો પાડે છે. બીજી બાજુ, ડિઝાઇન વ્યૂ વધુ જટિલ ક્વેરી લખવા માટે એક કાર્યક્ષમ પદ્ધતિ પૂરી પાડે છે, જેના દ્વારા ઉપયોગી માહિતી પુનઃપ્રાપ્ત કરવા માટે મલ્ટીપલ ટેબલને જોડી શકાય છે.

## સ્વાધ્યાય

1. લિબ્રેઓફીસ બેઝમાં ક્વેરીનો ઉપયોગ કરીને ટેબલમાંથી ચોક્કસ રેકોર્ડ પુનઃપ્રાપ્ત કરવાની પ્રક્રિયા સમજાવો.
2. મલ્ટીપલ ટેબલમાંથી ડેટા કેવી રીતે પુનઃપ્રાપ્ત કરી શકાય તે વર્ણવો?
3. ડિઝાઇન વ્યૂનો ઉપયોગ કરીને ક્વેરી કેવી રીતે બનાવી શકાય તે સમજાવો.
4. લિબ્રેઓફીસ બેઝમાં ક્વેરીનો હેતુ શું છે?
5. સિમ્પલ ક્વેરી અને પેરામીટર ક્વેરી વચ્ચેનો તફાવત સમજાવો.
6. લિબ્રેઓફીસ બેઝમાં કયો વ્યૂ ક્વેરીને વિઝ્યુઅલ રીતે બનાવવા માટે GUI પ્રદાન કરે છે?
7. Students નામના ટેબલમાંથી બધા ફીલ્ડ સિલેક્ટ કરવા માટેનું SQL સ્ટેટમેન્ટ લખો.
8. આ ક્વેરીનું આઉટપુટ શું છે?  
*SELECT "Name", "Age" FROM "Employees" WHERE "Age" > 30;*
9. ક્વેરીમાં વાઈલ્ડકાર્ડનો ઉપયોગ શું છે?
10. ક્વેરી લખવામાં કાર્ડિનલિટીનો ઉપયોગ શું છે?
11. સાચું કે ખોટું જણાવો.
  - (1) લિબ્રેઓફીસ બેઝમાં ક્વેરી ફક્ત એક જ ટેબલ માંથી ડેટા પુનઃપ્રાપ્ત કરી શકે છે.
  - (2) આપણે લિબ્રેઓફીસ બેઝમાં ડેટા પુનઃપ્રાપ્ત કરવા માટે SQL સ્ટેટમેન્ટ્સનો ઉપયોગ કરી શકીએ છીએ.
  - (3) પેરામીટર ક્વેરી યુઝરને રનટાઈમમાં વેલ્યુ એન્ટર કરીને ડેટાને ફિલ્ટર કરવાની મંજૂરી આપે છે.
  - (4) ક્વેરીમાં ડેટા પુનઃપ્રાપ્ત કરતી વખતે પેટર્ન મેચિંગ માટે % જેવા વાઈલ્ડકાર્ડ કેરેક્ટરનો ઉપયોગ કરી શકાય છે.
  - (5) લિબ્રેઓફીસ બેઝમાં ક્વેરીને સેવ કરી શકાય છે અને તેનો ઉપયોગ વારંવાર કરી શકાય છે.



## 12. ખાલી જગ્યા પૂરો.

- (1) લિબ્રેઓફિસ બેઝમાં એક અથવા વધુ ટેબલમાંથી ચોક્કસ માહિતી પ્રાપ્ત કરવા માટે \_\_\_\_\_ નો ઉપયોગ થાય છે.
- (2) લિબ્રેઓફિસ બેઝમાં \_\_\_\_\_ વ્યૂ યુઝરને SQL કોડ લખ્યા વિના વિઝ્યુઅલ રીતે ક્વેરી બનાવવાની સવલત આપે છે.
- (3) પેરામીટર ક્વેરીમાં, યુઝર ઈનપુટ \_\_\_\_\_ કૌંસનો ઉપયોગ કરીને પૂછવામાં આવે છે.
- (4) SQLમાં કોઈપણ અક્ષરોના ક્રમને મેચ કરવા માટે વાઈલ્ડકાર્ડ કેરેક્ટર \_\_\_\_\_ નો ઉપયોગ થાય છે.
- (5) ક્વેરીના પરિણામો \_\_\_\_\_ની જેમ ટેબલ ફોર્મેટમાં ડિસ્પ્લે કરી શકાય છે.

## 13. બહુવિકલ્પી પ્રશ્નો. સૌથી યોગ્ય જવાબ પસંદ કરો.

- (1) ડેટાબેઝમાંથી ચોક્કસ માહિતી પુનઃપ્રાપ્ત કરવા માટે લિબ્રેઓફિસ બેઝમાં કયા ઓબ્જેક્ટનો ઉપયોગ થાય છે?  
(a) ટેબલ (Table)      (b) ફોર્મ (Form)      (c) ક્વેરી (Query)      (d) રીપોર્ટ (Report)
- (2) SQL ક્વેરીમાં \* સિમ્બોલ શું દર્શાવે છે?  
(a) તમામ રો      (b) તમામ કોલમ      (c) તમામ ટેબલ      (d) પ્રાઈમરી કી
- (3) લિબ્રેઓફિસ બેઝમાં નીચેનામાંથી કયો વ્યૂ ક્વેરીને ગ્રાફિકલ રીતે બનાવવાની સવલત આપે છે?  
(a) SQL વ્યૂ      (b) ફોર્મ વ્યૂ      (c) ડીઝાઈન વ્યૂ      (d) ટેબલ વ્યૂ
- (4) પેરામીટર ક્વેરીમાં, યુઝરને ઈનપુટ માટે પ્રોમ્પ્ટ કરવા માટે શું વપરાય છે?  
(a) છગડિયો કૌંસ {}      (b) કૌંસ ()  
(c) મોટો કૌંસ []      (d) અવતરણ ચિહ્ન “ ”
- (5) લિબ્રેઓફિસ બેઝમાં ડેટા પુનઃપ્રાપ્તિ માટે નીચેનામાંથી કોનો ઉપયોગ થાય છે?  
(a) ફોર્મ      (b) ક્વેરી      (c) રીલેશન      (d) ટેબલ ડીઝાઈન
- (6) લિબ્રેઓફિસ બેઝમાં કયા વાઈલ્ડકાર્ડ કેરેક્ટર અક્ષરોના કોઈપણ ક્રમ સાથે મેચ થાય છે?  
(a) -      (b) #      (c) \*      (d) ?
- (7) ડેટાબેઝમાંથી ડેટા મેળવવા માટે કયા SQL કમાન્ડનો ઉપયોગ થાય છે?  
(a) UPDATE      (b) INSERT      (c) DELETE      (d) SELECT
- (8) 'A' અક્ષરથી શરૂ થતા નામો મેળવવા માટે, નીચેનામાંથી શેનો ઉપયોગ કરવામાં આવશે?  
(a) LIKE 'A\_'      (b) LIKE '\_A%'      (c) LIKE 'A\*'      (d) LIKE '%A'
- (9) SELECT "City" FROM "Customers" WHERE "Country" = 'India' આ ક્વેરી શું પરિણામ આપશે?  
(a) બધા દેશોના શહેરો      (b) ભારતમાં રહેલા ગ્રાહકોના શહેરો  
(c) બધા ગ્રાહકોના દેશો      (d) ભારતમાં રહેલા ગ્રાહકોના નામો



(10) જો ક્વેરી કોઈ રો પરિણામ સ્વરૂપે ડિસ્પ્લે ન કરે તો લિબ્રેઓફીસ બેઝ શું ડિસ્પ્લે કરશે?

- (a) એરર મેસેજ (b) NULL  
(c) ખાલી પરિણામની ગ્રીડ (d) 0

### પ્રાયોગિક સ્વાધ્યાય

1. નીચેના ટેબલમાં બતાવ્યા પ્રમાણે BookList નામનું ડેટાબેઝ ટેબલ બનાવો.

| BookID | BookTitle                         | Author                 | Genre            | Ratings |
|--------|-----------------------------------|------------------------|------------------|---------|
| 1      | The Hobbit                        | J. R. R. Tolkien       | Adventure        | 4       |
| 2      | The Adventures of Sherlock Holmes | Sir Arthur Conan Doyle | Suspense         | 5       |
| 3      | Oliver Twist                      | Charles Dickens        | Drama            | 3       |
| 4      | Adventures of huckleberry finn    | Mark Twain             | Children Fiction | 4       |

2. ક્વેરી વિઝાર્ડનો ઉપયોગ કરીને, સ્વાધ્યાય 1 માં બનાવેલ ટેબલ BookListમાંથી મૂળાક્ષરોના ક્રમમાં ગોઠવાયેલા પુસ્તકના નામ મેળવવા માટે ક્વેરી બનાવો.
3. ડિઝાઈન વ્યૂનો ઉપયોગ કરીને, ઓછામાં ઓછા 4 રેટિંગ ધરાવતા પુસ્તકોની યાદી ડિસ્પ્લે કરવા માટે ક્વેરી બનાવો.
4. "Genre" પેરામીટર સાથે ક્વેરી બનાવો, જેથી યુઝર આ પ્રકાર માટેના પુસ્તકો શોધી શકે.

