DROP table if exists WorksInsurance cascade;

DROP table if exists Themes cascade;

DROP table if exists SponsorExhibitions cascade;

DROP table if exists WorkTransaction cascade;

DROP table if exists ExhibitionLocations cascade;

DROP table if exists ExhibitionWorks cascade;

DROP table if exists WorkOwners cascade;

DROP table if exists WorksMedium cascade;

DROP table if exists Works cascade;

DROP table if exists Owners cascade;

DROP table if exists Locations cascade;

DROP table if exists Exhibitions cascade;

DROP table if exists Doors cascade;

DROP table if exists WorkLocations cascade;

--Need to create a simple data attribute for phone numbers. If you don’t have them, don’t worry!

--CREATE DOMAIN PhoneNumber as varchar(13);

--ALTER TABLE owners add column ownphonenumber PhoneNumber;

CREATE TABLE Owners (

ownName OwnerName NOT NULL,

ownEmail OwnerEmail,

ownCity OwnerCity,

ownState OwnerState,

ownStreetName OwnerStreetName,

ownBuildingNumber OwnerBuildingNumber,

ownisPartnerMuseum IsPartnerMuseum NOT NULL,

PRIMARY KEY(ownName)

);

--

CREATE TABLE Works (

worWorkCharID WorkCharID NOT NULL,

worWorkNumID WorkNumID NOT NULL,

worDatabaseEntryMuseum OwnerName NOT NULL,

worWorkName WorkName,

worWorkDescription WorkDescription,

worWorkPhysicalProperty WorkPhysicalProperty,

worWorkClassification WorkClassification,

worWorkCreator WorkCreator,

worWorkCreationDate WorkCreationDate,

worWorkBorrowable WorkBorrowable NOT NULL,

worWorkGeographicRegionOfOrigin GeographicRegionOfOrigin,

worWorkCountryOfOrigin CountryOfOrigin,

worWorkFieldOfScience FieldOfScience, --coordinate name

PRIMARY KEY(worWorkCharID, worWorkNumID, worDatabaseEntryMuseum),

FOREIGN KEY(worDatabaseEntryMuseum) REFERENCES Owners(ownName) ON UPDATE CASCADE

);

CREATE TABLE WorksMedium (

wmeWorkCharID WorkCharID NOT NULL,

wmeWorkNumID WorkNumID NOT NULL,

wmeDatabaseEntryMuseum OwnerName NOT NULL,

wmeWorkMedium WorkMedium NOT NULL,

PRIMARY KEY(wmeWorkCharID, wmeWorkNumID, wmeDatabaseEntryMuseum, wmeWorkMedium),

FOREIGN KEY(wmeWorkCharID, wmeWorkNumID, wmeDatabaseEntryMuseum) REFERENCES Works(worWorkCharID, worWorkNumID, worDatabaseEntryMuseum) ON UPDATE CASCADE

);

--Call locBuildingName locOwnerName (building name too vague)

CREATE TABLE Locations (

locLocationName LocationName, --was RoomName but could be towns

locBuildingName OwnerName, --will have to be owner name and the name building name should change

locIsTravellingExhibitionLocation IsTravellingExhibitionLocation,

locLocationSuggestedCapacityMin LocationSuggestedCapacity,

locLocationSuggestedCapacityMax LocationSuggestedCapacity,

locLocationMeasurementWidth LocationMeasurement,

locLocationMeasurementLength LocationMeasurement,

FOREIGN KEY(locBuildingName) REFERENCES Owners(ownName),

PRIMARY KEY(locLocationName, locBuildingName),

CHECK(locLocationSuggestedCapacityMin < locLocationSuggestedCapacityMax)

);

CREATE TABLE WorkLocations (

wolWorkCharID WorkCharID NOT NULL,

wolWorkNumID WorkNumID NOT NULL,

wolWorkDatabaseEntryMuseum OwnerName NOT NULL,

wolLocationName LocationName NOT NULL,

wolMuseumName OwnerName NOT NULL,

wolWorkTimeArrival WorkTime NOT NULL,

wolWorkTimeDeparture WorkTime,

Foreign Key (wolWorkCharID, wolWorkNumID, wolWorkDatabaseEntryMuseum) REFERENCES Works(worWorkCharID, worWorkNumID, worDatabaseEntryMuseum),

Foreign Key (wolLocationName, wolMuseumName) REFERENCES Locations(locLocationName, locBuildingName),

PRIMARY KEY(wolWorkCharID, wolWorkNumID, wolWorkDatabaseEntryMuseum, wolLocationName, wolMuseumName,wolWorkTimeArrival),

CHECK ((wolWorkTimeArrival<=wolWorkTimeDeparture) OR (wolWorkTimeDeparture is NULL))

);

CREATE TABLE Doors (

dorMuseumNameOrigin OwnerName NOT NULL,

dorLocationNameOrigin LocationName NOT NULL,

dorMuseumNameDestination OwnerName NOT NULL,

dorLocationNameDestination LocationName NOT NULL,

FOREIGN KEY(dorMuseumNameOrigin, dorLocationNameOrigin) REFERENCES Locations(locBuildingName, locLocationName),

FOREIGN KEY(dorMuseumNameDestination, dorLocationNameDestination) REFERENCES Locations(locBuildingName, locLocationName),

PRIMARY KEY(dorMuseumNameOrigin, dorLocationNameOrigin, dorMuseumNameDestination,dorLocationNameDestination)

);

--Missing Descriptions!!!

CREATE TABLE Exhibitions (

exhExhibitName ExhibitName NOT NULL,

exhMuseumName OwnerName NOT NULL,

exhExhibitDateStart ExhibitDate NOT NULL,

exhExhibitDateEnd ExhibitDate NOT NULL,

exhExhibitDescription ExhibitDescription, -- Added description

exhSecurityPersonName SecurityName,

exhIsTravellingExhibition IsTravellingExhibition,

PRIMARY KEY(exhExhibitName, exhMuseumName, exhExhibitDateStart),

FOREIGN KEY(exhMuseumName) REFERENCES Owners(OwnName),

CHECK( exhExhibitDateStart<exhExhibitDateEnd)

);

CREATE TABLE ExhibitionLocations (

eloExhibitName ExhibitName NOT NULL,

eloExhibitDateStart ExhibitDate NOT NULL,

eloLocationName LocationName NOT NULL,

eloMuseumOfLocation OwnerName NOT NULL,

eloMuseumOfExhibition OwnerName NOT NULL,

eloLocationDateStart LocationDate NOT NULL,

eloLocationDateEnd LocationDate,

PRIMARY KEY(eloExhibitName, eloExhibitDateStart, eloLocationName, eloMuseumOfLocation, eloLocationDateStart),

FOREIGN KEY(eloExhibitname, eloExhibitDateStart,eloMuseumOfExhibition) REFERENCES Exhibitions(exhExhibitName, exhExhibitDateStart, exhMuseumName),

FOREIGN KEY(eloLocationName, eloMuseumOfLocation) REFERENCES Locations(locLocationName, locBuildingName),

CHECK(((eloLocationDateEnd is NULL) OR (eloLocationDateStart<eloLocationDateEnd))

AND (eloExhibitDateStart<=eloLocationDateStart))

);

CREATE TABLE ExhibitionWorks (

exwExhibitName ExhibitName NOT NULL,

exwExhibitDateStart ExhibitDate NOT NULL,

exwMuseumOfExhibition OwnerName NOT NULL,

exwWorkCharID WorkCharID NOT NULL,

exwWorkNumID WorkNumID NOT NULL,

exwWorkDataBaseEntryMuseum OwnerName NOT NULL,

exwDateWorkAdded DateWorkAdded NOT NULL,

exwDateWorkRemoved DateWorkRemoved,

PRIMARY KEY(exwExhibitname, exwExhibitDateStart, exwMuseumOfExhibition, exwWorkCharID, exwWorkNumID, exwWorkDataBaseEntryMuseum, exwDateWorkAdded),

FOREIGN KEY(exwExhibitName, exwExhibitDateStart,exwMuseumOfExhibition) REFERENCES Exhibitions(exhExhibitName,exhExhibitDateStart, exhMuseumName),

FOREIGN KEY(exwWorkCharID, exwWorkNumID, exwWorkDatabaseEntryMuseum) REFERENCES Works(worWorkCharID, worWorkNumID, worDatabaseEntryMuseum),

CHECK((exwDateWorkAdded<exwDateWorkRemoved) OR (exwDateWorkRemoved is NULL))

);

--Change EntryLocation to EntryMuseum?

CREATE TABLE WorkOwners (

wonWorkCharID WorkCharID NOT NULL,

wonWorkNumID WorkNumID NOT NULL,

wonWorkDatabaseEntryLocation OwnerName NOT NULL,

wonOwnerName OwnerName NOT NULL,

wonOwnershipDateStart OwnershipDate NOT NULL,

wonOwnershipDateEnd OwnershipDate,

PRIMARY KEY(wonWorkCharID, wonWorkNumId, wonworkDatabaseEntryLocation, wonOwnerName, wonOwnershipDateStart),

FOREIGN KEY(wonWorkCharID, wonWorkNumID, wonWorkDatabaseEntryLocation) REFERENCES Works(worWorkCharID, worWorkNumID, worDatabaseEntryMuseum),

FOREIGN KEY(wonOwnerName) REFERENCES Owners(ownName),

CHECK((wonOwnershipDateStart<wonOwnershipDateEnd) OR (wonOwnershipDateEnd is NULL))

);

CREATE TABLE WorkTransaction (

wtrWorkCharID WorkCharID NOT NULL,

wtrWorkNumID WorkNumID NOT NULL,

wtrWorkDatabaseEntryMuseum OwnerName NOT NULL,

wtrMuseumInvolved OwnerName NOT NULL,

wtrClient OwnerName,

wtrTransactiontype TransactionType NOT NULL,

wtrTransactionTime TransactionTime NOT NULL,

PRIMARY KEY(wtrWorkCharID, wtrWorkNumID, wtrWorkDatabaseEntryMuseum, wtrMuseumInvolved, wtrTransactionType, wtrTransactionTime),

FOREIGN KEY(wtrWorkCharID, wtrWorkNumID, wtrWorkDatabaseEntryMuseum) REFERENCES Works(worWorkCharID, worWorkNumID, worDatabaseEntryMuseum),

FOREIGN KEY(wtrWorkDatabaseEntryMuseum) REFERENCES Owners(ownName),

FOREIGN KEY(wtrClient) REFERENCES Owners(ownName)

);

CREATE TABLE SponsorExhibitions (

speSponsorName SponsorName NOT NULL,

speExhibitName ExhibitName NOT NULL,

speMuseumName OwnerName NOT NULL,

speExhibitDateStart ExhibitDate NOT NULL,

speSponsorAmount SponsorAmount,

PRIMARY KEY(speSponsorName, speExhibitName, speMuseumName, speExhibitDateStart),

FOREIGN KEY(speExhibitName, speMuseumName, speExhibitDateStart) REFERENCES Exhibitions(exhExhibitName, exhMuseumName, exhExhibitDateStart)

);

--Change EntryLocation to EntryMuseum

CREATE TABLE Themes (

wthWorkCharID WorkCharID NOT NULL,

wthWorkNumID WorkNumID NOT NULL,

wthWorkDatabaseEntryLocation OwnerName NOT NULL,

wthTheme Theme NOT NULL,

PRIMARY KEY(wthWorkCharID, wthWorkNumID, wthWorkDatabaseEntryLocation, wthTheme),

FOREIGN KEY(wthWorkCharID, wthWorkNumID, wthWorkDatabaseEntryLocation) REFERENCES Works(worWorkCharID, worWorkNumID, worDatabaseEntryMuseum)

);

--Change EntryLocation to EntryMuseum?

CREATE TABLE WorksInsurance (

winWorkCharID WorkCharID NOT NULL,

winWorkNumID WorkNumID NOT NULL,

winWorkDatabaseEntryLocation OwnerName NOT NULL,

winWorkInsureValue WorkInsureValue NOT NULL,

winInsureDateStart InsureDate NOT NULL,

winInsureDateEnd InsureDate,

PRIMARY KEY(winWorkCharID, winWorkNumID, winWorkDatabaseEntryLocation, winInsureDateStart),

FOREIGN KEY(winWorkCharID, winWorkNumID, winWorkDatabaseEntryLocation) REFERENCES Works(worWorkCharID, worWorkNumID, worDatabaseEntryMuseum),

CHECK((winInsureDateStart< winInsureDateEnd) OR (winInsureDateEnd is NULL))

);

--We can add this at merge time

--RoomCondition

--Table