## **Project 3: Class Diagram**

Team Members: Ethan Brenner Kyle Kappes-Sum Malena Schoeni

## unit: String) accesses data to populate dropdown menus. calls helper methods. + attribute1:type = defaultValue + m catMenu: Element + m\_unitAMenu: Element + attribute2:type - attribute3:type + m unitBMenu: Element + m constMenu: Element + m constUnitMenu: Element + jouleToErg(joules: Number) + m\_formulaMenu: Element + ergToJ(btu: Number) + jouleToBtu(joules: Number) + m\_unitAInput: Element + m unitBOutput: Element + UNIV GAS CONST: Object + btuToJ(btu: Number) + m constOutput: Element + RHO OF WATER: Object + jouleToCal(joules: Number) + m formulaFields: Element + MU OF WATER: Object + calToJ(cal: Number) + NU\_OF\_WATER: Object + CAP\_MU\_OF\_WATER: Object + SIGMA\_OF\_WATER: Object + jouleToEV(joules: Number) + eVToJ(eV: Number) + constructor() + jouleToFootPoundForce(joules: Number) + initialize() + RHO OF AIR: Object + ftlbfToJ(fpf: Number) + populateCategories() + MU\_OF\_AIR: Object calls conversion methods accesses constant values + jouleToHorsepowerHour(joules: Number) + populateEnergyMenus() + NU\_OF\_AIR: Object + HEAT\_CAPACITY\_OF\_AIR: Object + hphToJ(hph: Number) + populatePressureMenus() + jouleTokWh(joules: Number) + populateNextDropdown(menuAld: String, menuBld: String) + M OF AIR: Object + kwhToJ(kwh: Number) + populateConstants() + K\_OF\_AIR: Object + showHelpText(ID: String) + jouleTokWs(joules: Number) + MOLEC\_GAS\_WEIGHTS: Object + kwsToJ(kws: Number) + hideHelpText(ID: String) + GRAVITY: Object + atmoTokPa(atmos: Number) + genConversionID(category: String, unitA: String, unitB: String) + kPaToAtmo(kPas: Number) + convertEnergy(value: Number, conversionID: String) + atmoToPa(atmos: Number) + convertPressure(value: Number, conversionID: String) + convert(category: String, unitA: String, unitB: String, value: Number) + paToAtmo(pas: Number) + convertHandler() + atmoToBar(atmos: Number) + barToAtmo(bars: Number) + categoryChange() + atmoToPoundPerSquaredInch(atmos: Number) + constChange() + poundPerSquaredInchToAtmo(lbfs: Number) + constHandler() + atmoToFootWater(atmos: Number) + calculateHandler() + footWaterToAtmo(footWaters: Number) + calcPVNRT(unknown: String) + atmoToMercInch(atmos: Number) + MercInchToAtmo(mercInches: Number) + atmoToMercMM(atmos: Number) calls formula calculation methods + mercMMToAtmo(mercMMs: Number) + atmoToKgf(atmos: Number) + kgfToAtmo(kgfs: Number) + atmoToMeterWater(atmos: Number) + meterWaterToAtmo(meterWaters: Number) FormulasSol + atmoToTorr(atmos: Number) + uConst: UnitConstant accesses constant values + torrToAtmo(torrs: Number) + fAbs: FormulaAbs + atmoToPsi(atmos: Number) + psiToAtmo(psis: Number) + pvNRT(obj: Number) + atmoToInchWater(atmos: Number) + inchWaterToAtmo(inchWaters: Number) + atmoToDynePerCM(atmos: Number) calls abstract calculation methods + dynePerCMToAtmo(dynes: Number)

FormulasAbs

+ objectSize(obj: Object) + findVar(obj: Object, arr: Array)

+ multiplySolve(numer: Array, denom: Array)

Config

+ CATEGORIES: Array

+ ENERGY\_UNITS: Array + PRESSURE UNITS: Array + CONSTANTS: Array

+ RHO\_OF\_WATER:Array + MU\_OF\_WATER:Array + NU\_OF\_WATER:Array

+ UNIV GAS CONSTANT: Array

+ CAP MU OF WATER:Array + SIGMA OF WATER:Array + FORMULAS:Array

+ getIndexOf(category: String,

UnitConstant