fpstracker.core BaseSampler Package :: fpstracker.core parent : PApplet samplingSize : int sampleList : List<Object> minSample : Object TrackerType : enum < < Interface > > maxSample : Object Sampling Package :: fpstracker.core lastSample : Object play() : void play : boolean MILLIS pause(): void BaseSampler(parent : PApplet, samplingSize : int, type : TrackerType) : void MEMORY startSampling(sample : int) : void play() : void CUSTOM endSampling(sample : int) : void pause() : void startSampling(sample : float) : void startSampling(sample : int) : void FLOAT endSampling(sample : float) : void endSampling(sample : int) : void computePanel(): void startSampling(sample : float) : void displayPanel(canvas : PGraphics, x : int, y : int) : void endSampling(sample : float) : void setSamplingSize(samplingSize : int) : void setSamplingSize(samplingSize : int) : void setWidth(width: int): void getSamplingSize(): int Γ − − − − − − − − − − − − − − − | setHeight(height : int) : void setSize(width : int, height : int) : void getSampleList() : List<Object>
getMinSample() : Object getSamplingSize(): int getMaxSample() : Object getSampleList() : List<int>
getMinSample() : int isPlaying(): boolean toString(): String BaseCustomTracker getMaxSample(): int Package :: fpstracker.core getSampleList() : List<float> getMinSample() : float parent : PApplet getMaxSample() : float sampler : Sampler fpstracker.ui panel : UIPanel isPlaying() : boolean getCanvas() : PGraphics BaseCustomTracker(parent : PApplet, samplingSlze : int, type : TrackerType) : void setColor(color : int) : void toString():String BaseUIPanel package :: fpstracker.ui Sampler SamplerMemory parent : PApplet Package :: fpstracker.core Package :: fpstracker.core canvas : PGraphics freeMemory: int width : int totalMemory : int height : int maxMemorySize : int color : int getMaxMemorySize(): int type: string getTotalMemory(): int sampler : Sampler getFreeMemory(): int BaseUIPanel(parent : PApplet, width : int, height : int) : void computeCanvas() : void display(context : PGraphics, x : int, y : int) getCanvas() : PGraphics setColor(color: int): void setWidth(width:int):void setHeight(height : int) : void CustomTracker setSize(width: int, height: int): void Stats Package :: fpstracker.core Package :: fpstracker.core parent : PApplet parent : PApplet sampler : Sampler actualPanel: int panel : UIPanel samplerList : List<BaseCustomTracker> UIPanel Tracker(parent : PApplet, samplingSize : int) : void package :: fpstracker.ui pre() : void draw() : void mouseEvent() : void displayNextPanel(): void **FPSTracker** displayPreviousPanel(): void Package :: fpstracker.core **UIMemoryPanel** startSampling(): void package :: fpstracker.ui parent : PApplet endSampling() : void sampler : Sampler computePanels : void panel : UIPanel displayPanels(canvas : PGraphics, x : int, y : int) : void play() : void pause() : void computeCanvas(canvas PGraphics, x : int, y : int) : void play(type : TrackerType) : void pause(type : TrackerType) : void MillisTracker setSamplingSize(samplingSize : int) : void Package :: fpstracker.core setWidth(width : int) : void setHeight(height : int) : void parent : PApplet setSize(width : int, height : int) sampler : Sampler getSamplingSize() : int getSamplingList(type : TrackerType) : List<int> panel : UIPanel getMinSample(type : TrackerType) : int getMaxSample(type : TrackerType) : int getMaxMemorySize() : int MemoryTracker getTotalMemory(): int Package :: fpstracker.core getFreeMemory() : int parent : PApplet sampler : Sampler panel : UIPanel