In [1]:
```
using Distributions
using Gadfly
using Interact
```

In [2]:
```
set_default_plot_size(25cm, 12cm)
```

In [4]:
```
@manipulate for samples in 10:50:1000, observations in 10:100:1000, β0 in 0:10, β1 in 0:0.1:1, u_μ in -5:5, u_σ in 1:5

    # Create lists for values of β_0 and β_1
    β0_hat = []
    β1_hat = []

    # Collect values of β among different samples
    for sample in 1:1:samples

        ## Initial
        x_values = rand(Uniform(0,100),observations)
        u = rand(Normal(u_μ, u_σ),observations)
        y_values = β0 + β1 * x_values + u

        ## Classical Linear Regression
        X = [ones(x_values) x_values]
        β = inv(X'*X)*X'*y_values

        ## Save values
        push!(β0_hat, β[1])
        push!(β1_hat, β[2])

    end

    # Plot

    ## Example values
    x_values = rand(Uniform(0,100),observations)
    u = rand(Normal(u_μ, u_σ),observations)
    y_values = β0 + β1 * x_values + u
    X = [ones(x_values) x_values]
    β = inv(X'*X)*X'*y_values

    # CLR graph
    graph_CLR = plot(
        layer(x=x_values, y=y_values, Geom.point, order=1),
        layer(x=x_values, y=X*β, Geom.line, order=2, Theme(default_color=colorant"black", line_width=1pt)),
        Guide.XLabel("Values of x"),
        Guide.YLabel("Values of y"),
        Guide.Title("Classical Linear Regression"),
        Coord.Cartesian(xmin=0,xmax=100,ymin=0,ymax=100, fixed=true)
    )

    # Distribtuion of β_1
    graph_β = plot(
        layer(x=β1_hat, Geom.histogram(bincount=50), order=1),
        layer(xintercept=[β1], Geom.vline(color=["black"], style=[[1mm]], size=[.5mm]), order=3),
        Guide.XLabel("Estimates of β_1"),
        Guide.YLabel("Number of Observations"),
        Guide.Title("Distribution of β_1")

    )

    hstack(graph_CLR, graph_β)

end
```

samples 460

observations 410

β0 5

β1 0.5

u_μ 0

u_σ 3

Out[4]: