var question = [“수도”,”인기대학”,”7+21”] var currentIndex = 0

@IBOutlet weak var questionLabel : UILabel!

override func viewDidLoad() { questionLabel.text = questions[currentIndex] }

@IBAction func nextQuestion(\_ sender: UIButton){ currentIndex = (currentIndex+1)%questions.count

questionLabel.text = questions[currentIndex]}

@IBOutlet weak var fahrenheitTextField: UITextField!

@IBOutlet weak var celsiusLabel: UILabel!

let tapGesture = UITapGestureRecognizer(target: self, action: #selector(dismissKeyboard))

view.addGestureRecognizer(tapGesture)

fahrenheitTextField.delegate = self

fahrenheitTextField = view.subviews[0] as! UITextField

celsiusLabel = view.subviews[4] as? UILabel

fahrenheitTextField.text = "123"

let celsiusValue = 5.0 / 9.0 \* (123.0 - 32)

celsiusLabel?.text = String(celsiusValue)

@IBAction func fahrenheitEditingChanged(\_ sender: UITextField) {

if let text = sender.text { // 그내용이 뭔가 있으면, 옵셔널 바인딩

if let fahrenheitValue = Double(text){

let celsiusValue = 5.0/9.0\*(fahrenheitValue - 32.0) // 변환

celsiusLabel.text = String(format: "%.2f", celsiusValue)}else{

celsiusLabel.text = "???"}}}

extension ConversionViewController{

@objc func dismissKeyboard(sender: UITapGestureRecognizer){

fahrenheitTextField.resignFirstResponder()}}

extension ConversionViewController: UITextFieldDelegate{

func textField(\_ textField: UITextField, shouldChangeCharactersIn

range: NSRange, replacementString string: String) -> Bool {

let existing = textField.text?.range(of: ".")

let comming = string.range(of: ".")

if existing != nil, comming != nil {

return false}

return true}}

let helloLabel = UILabel(frame: CGRect(x: 100, y: 100, width: 200, height: 30))

helloLabel.text = "Hello, Autolayout"

helloLabel.font = UIFont.systemFont(ofSize: 30, weight: .bold)

view.addSubview(helloLabel)

helloLabel.translatesAutoresizingMaskIntoConstraints = false // true로 해보라

let centerXConstraint = helloLabel.centerXAnchor.constraint(equalTo: view.centerXAnchor, constant: 0)

let centerYConstraint = helloLabel.centerYAnchor.constraint(equalTo: view.centerYAnchor, constant: 0)

centerXConstraint.isActive = true

centerYConstraint.isActive = true

view.addSubview(helloLabel)

helloLabel.centerXAnchor.constraint(equalTo: view.safeAreaLayoutGuide.centerXAnchor).isActive = true

helloLabel.centerYAnchor.constraint(equalTo: view.safeAreaLayoutGuide.centerYAnchor).isActive = true

helloLabel.leadingAnchor.constraint(equalTo: view.safeAreaLayoutGuide.leadingAnchor, constant: 20).isActive = true

helloLabel.trailingAnchor.constraint(equalTo: view.safeAreaLayoutGuide.trailingAnchor, constant: -20).isActive = true

nameLabel.setContentHuggingPriority(.defaultHigh, for: .horizontal)

nameTextField.setContentHuggingPriority(.defaultLow, for: .horizontal)

nameLabel.centerYAnchor.constraint(equalTo: outer.centerYAnchor)

@IBOutlet weak var birdImageView: UIImageView!

@IBOutlet weak var birdTopConstraint: NSLayoutConstraint!

@IBOutlet weak var birdLeadingConstraint: NSLayoutConstraint!

var timer: Timer! var direction = 1

override func viewDidLoad() {

let button = UIButton()

button.setTitle("Start", for: .normal) button.backgroundColor = .green

button.translatesAutoresizingMaskIntoConstraints = false

button.addTarget(self, action: #selector(buttonPressed), for: .touchUpInside)

view.addSubview(button)

NSLayoutConstraint.activate([

button.centerXAnchor.constraint(equalTo: view.safeAreaLayoutGuide.centerXAnchor),

button.bottomAnchor.constraint(equalTo: view.safeAreaLayoutGuide.bottomAnchor),

button.widthAnchor.constraint(equalTo: view.widthAnchor, multiplier: 0.5)

])}

@objc func buttonPressed(\_ sender: UIButton){

if let text = sender.titleLabel?.text{

if text == "Start"{

timer = Timer.scheduledTimer(timeInterval: 0.1, target: self,selector: #selector(timerAction),userInfo: nil, repeats: true)}

else{timer.invalidate()// 타이머 중지 }

sender.setTitle((text == "Start") ? "Stop" : "Start", for: .normal) } }

@objc func timerAction(){// Action함수처럼 @objc가 필요함

birdLeadingConstraint.constant += CGFloat(direction\*10) // 왼쪽 여백의 값을 변경하여 새를 이동시킨다.

if birdLeadingConstraint.constant + birdImageView.frame.size.width >= view.frame.size.width{// 오른쪽 끝에 도달하면

direction = -1 // 새의 방향을 반전함

birdImageView.transform = .init(scaleX: -1, y: 1) }

else if birdLeadingConstraint.constant < 0{

direction = 1

birdImageView.transform = .init(scaleX: 1, y: 1) } }