@IBAction func uploadBtnDidTouch(sender: AnyObject) {

let actionSheet = UIAlertController(title: "Photo", message: nil, preferredStyle: .ActionSheet)

// show choices to the user so the user can choose from take photo, choose from library and cancel

actionSheet.addAction(UIAlertAction(title: "Take Photo", style: .Default, handler: {action in

self.takePhoto()

}))

actionSheet.addAction(UIAlertAction(title: "Choose From Library", style: .Default, handler: {action in

self.choosePhoto()

}))

actionSheet.addAction(UIAlertAction(title: "Cancel", style: .Cancel, handler: nil))

self.presentViewController(actionSheet, animated: true, completion: nil)

}

// function which allow the user to take photo

func takePhoto(){

let imagePickerController = UIImagePickerController()

if UIImagePickerController.availableCaptureModesForCameraDevice(.Rear) != nil{

imagePickerController.sourceType = .Camera

imagePickerController.cameraCaptureMode = .Photo

imagePickerController.delegate = self

presentViewController(imagePickerController, animated: true, completion: nil)

}

else{

noCamera()

}

}

// choose photo from library

func choosePhoto(){

let imagePickerController = UIImagePickerController()

// Only allow photos to be picked, not taken.

imagePickerController.sourceType = .PhotoLibrary

// Make sure ViewController is notified when the user picks an image.

imagePickerController.delegate = self

presentViewController(imagePickerController, animated: true, completion: nil)

}

// MARK: UIImagePickerControllerDelegate

func imagePickerControllerDidCancel(picker: UIImagePickerController) {

// Dismiss the picker if the user canceled.

dismissViewControllerAnimated(true, completion: nil)

}

func imagePickerController(picker: UIImagePickerController, didFinishPickingMediaWithInfo info: [String : AnyObject]) {

// The info dictionary contains multiple representations of the image, and this uses the original.

let selectedImage = info[UIImagePickerControllerOriginalImage] as! UIImage

// Set photoImageView to display the selected image.

imageView.image = selectedImage

// Dismiss the picker.

dismissViewControllerAnimated(true, completion: nil)

}

func noCamera(){

let alertVC = UIAlertController(

title: "No Camera",

message: "Sorry, this device has no camera",

preferredStyle: .Alert)

let okAction = UIAlertAction(

title: "OK",

style:.Default,

handler: nil)

alertVC.addAction(okAction)

presentViewController(

alertVC,

animated: true,

completion: nil)

}