

### Opción de Solo Lectura en Created At para Evitar Modificaciones

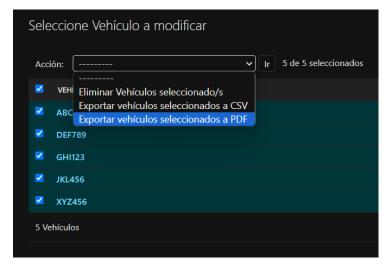
```
@admin.register(Vehicle)
class VehicleAdmin(admin.ModelAdmin):
    actions = ["export_to_csv", "export_to_pdf"]
    readonly_fields = ('created_at',)
```

```
class Vehicle(models.Model):
    registration_plate = models.CharField(max_length=6, primary_key=True, verbose_name="Placa de registro")
    brand = models.CharField(max_length=20, null=False, blank=False, verbose_name="Marca")
    model = models.CharField(max_length=20, null=False, blank=False, verbose_name="Modelo")
    year = models.DateField(null=False, blank=False, verbose_name="Ario")
    active = models.BooleanField(default=False, verbose_name="Activo")
    created_at = models.DateTimeField(default=timezone.now, verbose_name="Creado en")

def __str__(self):
    return self.registration_plate

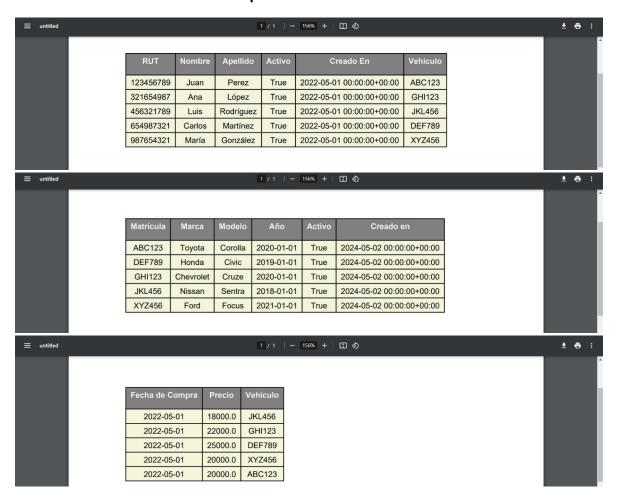
class Meta:
    verbose_name = "Vehículo"
    verbose_name_plural = "Vehículos"
    ordering = ["registration_plate"]
```

# Opción de Exportar en CSV y PDF



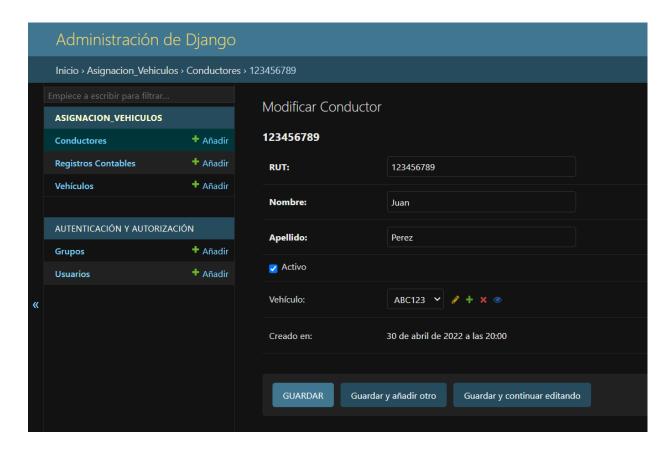
```
def export_to_pdf(self, request, queryset):
   response = HttpResponse(content_type="application/pdf")
   response["Content-Disposition"] = 'attachment; filename="Vehículos.pdf"'
   p = canvas.Canvas(response)
   data = [["Matrícula", "Marca", "Modelo", "Año", "Activo", "Creado en"]]
   for vehicle in queryset:
       data.append([
           vehicle.registration_plate,
           vehicle.brand,
           vehicle.model,
           vehicle.year,
           vehicle.active,
           str(vehicle.created_at), # Convertir a cadena si es necesario
   table = Table(data)
   style = TableStyle([
       ('BACKGROUND', (0, 0), (-1, 0), colors.grey),
       ('TEXTCOLOR', (0, 0), (-1, 0), colors.whitesmoke),
       ('ALIGN', (0, 0), (-1, -1), 'CENTER'),
       ('FONTNAME', (0, 0), (-1, 0), 'Helvetica-Bold'),
       ('BOTTOMPADDING', (0, 0), (-1, 0), 12),
       ('BACKGROUND', (0, 1), (-1, -1), colors.beige),
        ('GRID', (0, 0), (-1, -1), 1, colors.black)
   table.setStyle(style)
   table.wrapOn(p, 800, 600)
   table.drawOn(p, 72, 600)
   p.showPage()
   p.save()
   return response
export_to_csv.short_description = "Exportar vehículos seleccionados a CSV"
export_to_pdf.short_description = "Exportar vehículos seleccionados a PDF"
```

# **Exportar Archivo PDF**

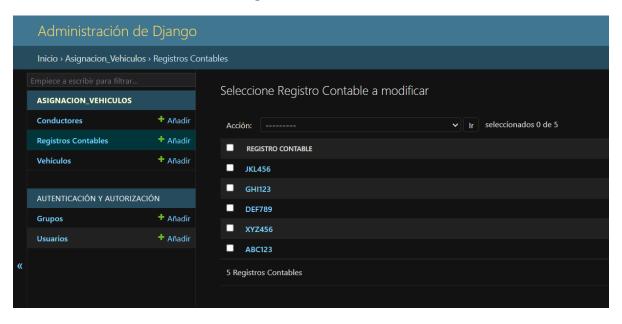


#### **Conductores**





# **Registros Contables**





# **Vehículos**



